U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

PB-264 185

Resources Allocation to Optimize Mining Pollution Control

Ohio State Univ Research Foundation, Columbus

Prepared for

Industrial Environmental Research Lab -Cincinnati, Ohio Extraction Technology Branch

Nov 76

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO.
EPA-600/2-76-112

2.

3. RECIPIENT'S ACCESSION NO.

4. TITLE AND SUBTITLE
RESOURCES ALLOCATION TO OPTIMIZE MINING POLLUTION CONTROL

5. REPORT DATE
November 1976 (Issuing Date)

6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S)
Kenesaw S. Shumate, E. E. Smith, Vincent T. Ricca, and Gordon M. Clark

8. PERFORMING ORGANIZATION REPORT NO.
RF 3484

9. PERFORMING ORGANIZATION NAME AND ADDRESS
The Ohio State University Research Foundation
1314 Kinnear Road
Columbus, Ohio 43212

10. PROGRAM ELEMENT NO.
EHE 623 05-01-08A-05

11. CONTRACT/GRANT NO.
68-01-0724

12. SPONSORING AGENCY NAME AND ADDRESS
Industrial Environmental Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency
Cincinnati, Ohio 45268

13. TYPE OF REPORT AND PERIOD COVERED
Final, 6/29/72 - 9/30/73

14. SPONSORING AGENCY CODE
EPA/600/12

15. SUPPLEMENTARY NOTES

16. ABSTRACT

A comprehensive model for mine drainage simulation and optimization of resource allocation to control mine acid pollution in a watershed has been developed. The model is capable of: (a) producing a time trace of acid load and flow from acid drainage sources as a function of climatic conditions; (b) generating continuous receiving stream flow data from precipitation data; (c) predicting acid load and flow from mine drainage sources using precipitation patterns and watershed status typical of "worst case" conditions that might be expected, e.g., once every 10 or 100 years; and (d) predicting optimum resource allocation using alternative methods of treatment and/or abatement for "worst case" conditions during both wet and dry portions of the hydrologic year. The model is comprehensive and may, therefore, be more detailed than required. This attention to detail was given in the belief that it will be easier to simplify the model than to modify it to increase detail. Because of the detail incorporated in the model as now constituted, a large amount of field data is required as input. In most cases, the desired field data are not now available. The model has not been fully tested or compared to real systems, nor has sensitivity to input data been determined. Therefore reliability of the model, and the necessity of detailed field data, have not been established. Comparisons with real systems are necessary to determine the level of simplification that can be permitted before the validity or usefulness of the model is impaired.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group
Strip mining, Coal mining, Mathematical models, Drainage, Watersheds, Refuse	Mine drainage, Computer models, Watershed models, Deep mines, Refuse piles, Coal mine drainage, Acid generation, Acid drainage treatment	08/I, 08/H 08/G, 08/M

18. DISTRIBUTION STATEMENT
Release to public

19. SECURITY CLASS *(This Report)*
Unclassified

20. SECURITY CLASS *(This page)*
Unclassified

21. NO. OF PAGES
496

22. PRICE

EPA Form 2220-1 (9-73)

1a

☆USGPO: 1977 — 757-056/5461 Region 5-II

PB 264 185

EPA-600/2-76-[illegible]
December 1976

Environmental Protection Technology Series

UNITED STATES ENVIRONMENTAL PROTECTION AGENCY

REPRODUCED BY
NATIONAL TECHNICAL
INFORMATION SERVICE
U. S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA. 22161

RESEARCH REPORTING SERIES

Research reports of the Office of Research and Development, U.S. Environmental Protection Agency, have been grouped into five series. These five broad categories were established to facilitate further development and application of environmental technology. Elimination of traditional grouping was consciously planned to foster technology transfer and a maximum interface in related fields. The five series are:

1. Environmental Health Effects Research
2. Environmental Protection Technology
3. Ecological Research
4. Environmental Monitoring
5. Socioeconomic Environmental Studies

This report has been assigned to the ENVIRONMENTAL PROTECTION TECHNOLOGY series. This series describes research performed to develop and demonstrate instrumentation, equipment, and methodology to repair or prevent environmental degradation from point and non-point sources of pollution. This work provides the new or improved technology required for the control and treatment of pollution sources to meet environmental quality standards.

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161.

EPA-600/2-76-112
November 1976

RESOURCES ALLOCATION TO OPTIMIZE MINING POLLUTION CONTROL

by

Kenesaw S. Shumate, E. E. Smith,
Vincent T. Ricca, Gordon M. Clark

The Ohio State University Research Foundation
Columbus, Ohio 43212

Contract No. 68-01-0724

Project Officer

Eugene F. Harris
Extraction Technology Branch
Industrial Environmental Research Laboratory
Cincinnati, Ohio 45268

INDUSTRIAL ENVIRONMENTAL RESEARCH LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
CINCINNATI, OHIO 45268

DISCLAIMER

This report has been reviewed by the Industrial Environmental Research Laboratory-Cincinnati, U.S. Environmental Protection Agency, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the U.S. Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

FOREWORD

When energy and material resources are extracted, processed, converted, and used, the pollution to our environment and to our aesthetic and physical well-being requires corrective approaches that recognize the complex environmental impact these operations have.

The Industrial Environmental Research Laboratory - Cincinnati uses a multidisciplinary approach to develop and demonstrate technologies what will rectify the pollutional aspects of these operations. The Laboratory assesses the environmental and socio-economic impact of industrial and energy-related activities and identifies, evaluates and demonstrates control alternatives.

This report presents a comprehensive model for mine drainage simulation. The model predicts pollution loads for given situations and then recommends optimum allocation of resources for treatment of abatement procedures. The work presented in this report is the first of several projects aimed at computer analysis of mine sites.

The model described herein has not been fully tested against a real situation. Other contracts are planned to continue this type of research. The product of this study, and following studies, will be of use to planning agencies and the mining industry. Through use of tools such as this, we may be able to increase production of vital energy resources while continuing to improve the environment.

David G. Stephan, Director
Industrial Environmental
Research Laboratory
Cincinnati

ABSTRACT

A comprehensive model for mine drainage simulation and optimization of resource allocation to control mine acid pollution in a watershed has been developed.

The model is capable of: (a) producing a time trace of acid load and flow from acid drainage sources as a function of climatic conditions; (b) generating continuous receiving stream flow data from precipitation data; (c) predicting acid load and flow from mine drainage sources using precipitation patterns and watershed status typical of "worst case" conditions that might be expected, e.g., once every 10 or 100 years; and (d) predicting optimum resource allocation using alternative methods of treatment and/or abatement for "worst case" conditions during both wet and dry portions of the hydrologic year.

The model is comprehensive and may, therefore, be more detailed than required. This attention to detail was given in the belief that it will be easier to simplify the model than to modify it to increase detail.

Because of the detail incorporated in the model as now constituted, a large amount of field data is required as input. In most cases, the desired field data are not now available.

The model has not been fully tested or compared to real systems, nor has sensitivity to input data been determined. Therefore reliability of the model, and the necessity of detailed field data, have not been established. Comparisons with real systems are necessary to determine the level of simplification that can be permitted before the validity or usefulness of the model is impaired.

This report was submitted in fulfillment of Contract Number 68-01-0724 by The Ohio State University Research Foundation under the sponsorship of the Environmental Protection Agency.

KEY WORDS:

Mine Drainage, Computer Models, Watershed Models, Deep Mines, Refuse Piles, Stripmines, Coal Mine Drainage, Acid Generation, Acid Drainage Treatment

CONTENTS

Page

Disclaimer ii
Foreword iii
Abstract iv
List of Figures vii
List of Tables xi
Acknowledgments xiii

Section

I CONCLUSIONS 1

II RECOMMENDATIONS 2

III INTRODUCTION 3

IV SHORT DISCUSSION OF THE PROJECT MODEL CONTENT AND OPERATION 5

Basic Physical Phenomena 5

- Hydrologic Model 5
  - Hydrologic Cycle and Its Model 6
  - History of the Hydrologic Model Development 6
  - Model Operation 8
  - Polluted Water Generated by Mining Activities 8
- Acid Generation Model 10
  - Pyrite Oxidation 10
  - Removal of Oxidation Products 12
- Pollutant Source Models 14
  - Deep Mine Source Model 19
  - Combined Refuse Pile-Strip Mine Source Model 19
- Basin Model 19

CONTENTS (continued)

Section		Page
	Cost-Effectiveness Model	21
	Acid Control Alternatives	22
	Socio-Political-Economic Considerations	24
	Basin Optimization Models	24
V	ILLUSTRATIVE EXAMPLES	33
	Source Models	33
	Input Requirements	34
	Basin Information	34
	Climatic Data	34
	Deep Mine Information	34
	Refuse Pile-Strip Mine Information	34
	Discharge Data	35
	Examples of Output	35
	Deep Mine Source Model	35
	Combined Refuse Pile-Strip Mine Source Model	37
	Other Information	40
	Basin Optimization	40
VI	EVALUATION OF OVERALL TECHNIQUE	49
	Source Model	50
	Optimization Model	54
VII	PUBLICATIONS	56
VIII	APPENDICES	57
	Appendix A - Deep Mine Pollutant Source Model	58
	Appendix B - The Refuse Pile and Strip Mine Pollutant Source Models	104
	Appendix C - Optimization Model for Resource Allocation to Abate Mine Drainage Pollution	183
	Appendix D - Cost of Drainage Treatment	459

LIST OF FIGURES

Figure No.		Page
1	Schematic of hydrologic cycle	7
2	Moisture accounting in Stanford Watershed Model	9
3	Idealized deep mine element	12
4	Schematic of Deep Mine Drainage Model	15
5	Schematic of Combined Refuse Pile-Strip Mine Model	16
6	The Deep Mine Source Model	17
7	Total acid load and acid load from each source area	18
8	Schematic diagram of subdivided basin	20
9	Input-output diagram for Basin Optimization Models	24
10	Stream network	25
11	Illustration of stream network with potential instream treatment facilities	27
12	Single stream with adjoining land uses	32
13	Streamflow hydrograph at the Big Four Watershed Outlet	35
14	Daily minewater discharge from McDaniels' Mine	36
15	Daily acid load discharge from McDaniels' Mine	36
16	Acid load nonuniform short duration rain	37
17	Stream flow hydrograph at the basin outlet	38
18	Daily flows from acid producing areas	38
19	Daily acid loads from acid producing areas	39

LIST OF FIGURES (continued)

Figure No. Page

20 Flow and acid load from acid producing areas - short duration rain 39

21 Stream network 41

A.1 Schematic of hydrologic cycle 61

A.2 Underground pyritic system 62

A.3 Moisture accounting in Stanford Watershed Model 65

A.4 Schematic of Acid Mine Drainage Model 68

A.5 Infiltration from UZS that is held in LZS 69

A.6 Logic diagram of The Ohio State University version of the Stanford Watershed Model 72

A.7 Logic diagram of the Acid Mine Drainage Model 75

A.8 The test watershed site 77

A.9 Double mass plot of precipitation data 78

A.10 Streamflow hydrograph at the Big Four Hollow watershed outlet 81

A.11 Daily minewater discharge from McDaniels' Mine 84

A.12 Daily acid load discharge from McDaniels' Mine 85

B.1 Hydrologic cycle on a refuse pile 108

B.2 Step diagram of the Refuse Pile Model 112

B.3 Acid load for continuous rain 135

B.4 Acid load short interval continuous rain 137

B.5 Acid load nonuniform short duration rain 138

B.6 Soil column 152

B.7 ACDPRO Program listing 157

B.8 ACDPRO flow chart 158

LIST OF FIGURES (continued)

Figure No.		Page
B.9	The ACDSEC Subroutine Program	162
B.10	The ACDSTR Subroutine Program	164
B.11	CRPSMM (Combined Refuse Pile - Strip Mine Model) block diagram	170
C.1	Single-stream watershed	184
C.2	Effect of instream treatment facilities	186
C.3	Single stream with adjoining land uses	192
C.4	Illustration of stream and node designation	197
C.5	Program MAXEF	259
C.6	Function CABAT	297
C.7	Function CSAV	300
C.8	Function EFFECT	304
C.9	Subroutine ERROR	309
C.10	Subroutine NEFESE	310
C.11	Subroutine TOFE	321
C.12	Subroutine TONE	324
C.13	Subroutine ZOE	329
C.14	Program ALCOT	340
C.15	Subroutine CONO	382
C.16	Subroutine ERROR	389
C.17	Subroutine NEXFES	390
C.18	Function NON	402
C.19	Subroutine PIMAX	404

LIST OF FIGURES (continued)

Figure No.		Page
C.20	Subroutine PTMX	411
C.21	Subroutine STORE	436
C.22	Function TCOST	437
C.23	Subroutine TOFF	442
C.24	Subroutine TON	448
C.25	Subroutine ZO	455
D.1	A plot of acidity concentration vs. lime cost	470
D.2	A plot of acidity concentration vs. lime cost 1967 West Virginia University data	473

LIST OF TABLES

Table No.		Page
1	Illustrative variation of basic value with maximum pollution concentration	30
2	Mine sources and pollution control costs	43
3	Average annual pollutant loads	44
4	Pollutant and stream flows spring rates	45
5	Pollutant and stream flows summer rates	46
6	Minimum cost (DWMC) model results	47
7	Maximum effectiveness (DWME) model results	47
8	Minimum annual pollution control costs from the DWMC model	48
A.1	SWM parameters	64
A.2	AMD parameters	67
A.3	Basin and mine characteristics	79
A.4	Daily infiltrating water reaching the mine aquifer	80
A.5	Synthesized daily minewater discharge	82
A.6	Synthesized daily acid load	83
A.7	Synthesized monthly and annual minewater discharge and acid load	86
B.1	Input variables obtained from the Stanford Watershed Model	113
B.2	Input variables into the refuse pile	114
B.3	Internal variables of the refuse pile model	115

LIST OF TABLES (continued)

Table No.		Page
B.4	Daily acid load in direct runoff	125
B.5	Daily acid load in interflow	126
B.6	Daily acid load in baseflow	127
B.7	Daily acid load	128
B.8	Daily direct runoff reaching receiving stream	129
B.9	Daily interflow reaching receiving stream	130
B.10	Daily baseflow reaching receiving stream	131
B.11	Daily flow reaching receiving stream	132
B.12	Monthly summary of acid loads and flows	133
B.13	Specific day output	134
B.14	ACDPRO Program input and output	160
B.15	CRPSMM internal variables	165
B.16	CRPSMM input variables	166
C.1	Illustrative variation of basic value with maximum pollution concentration	192
D.1	Cost and effectiveness of various techniques for controlling acid mine drainage	460
D.2	Estimation of capital investment cost	461
D.3	Estimation of total production costs	462
D.4	Estimated costs of lime neutralization of acid mine drainage	464
D.5	Estimated costs of lime neutralization of acid mine drainage based on equation (D.4)	466
D.6	Estimated costs of lime neutralization of acid mine drainage based on equation (D.4)	467

LIST OF TABLES (continued)

Table No.		Page
D.7	Estimated costs of lime neutralization of acid mine drainage based on equation (D.5)	468
D.8	Estimated costs of lime neutralization of acid mine drainage based on equation (D.5)	469

ACKNOWLEDGMENTS

The authors are grateful for the assistance of Dr. John T. Heibel, Assistant Professor of Chemical Engineering, The Ohio State University, for his efforts in assembling cost functions information used in the optimization model.

The following graduate students have made significant contributions in the research project and much of their writing has been incorporated into this report:

Kurtis Chow, Post Masters, Department of Chemical Engineering, The Ohio State University.

Stanley Johnson, Master of Science, Department of Civil Engineering, The Ohio State University.

Arthur Maupin, Master of Science, Department of Chemical Engineering, The Ohio State University.

Richard K. Law, Master of Science, Department of Chemical Engineering, The Ohio State University.

David Bitters, Ph. D. candidate and Graduate Research Associate, Department of Industrial and Systems Engineering, The Ohio State University.

Marilyn Vaughn, Master of Science, Department of Industrial and Systems Engineering, The Ohio State University.

The following personnel of the Environmental Protection Agency have contributed much effort in technical assistance and administration of this project.

Eugene F. Harris, Project Officer, Mining Pollution Control Branch, National Environmental Protection Agency, Cincinnati, Ohio.

Ronald D. Hill, Mining Pollution Control Branch, National Environmental Protection Agency, Cincinnati, Ohio.

The project investigators are grateful to The Ohio State University Research Foundation personnel for their administrative assistance and coordination with the sponsor.

Finally, a special thanks to Chet Ball and his staff for their fine job of typing and drafting performed in the writing of this report.

SECTION I

CONCLUSIONS

1. Source models for predicting the time dependent acid production from deep (underground) mines, refuse piles, and spoil banks have been developed.

2. The source models simulate flow and acid load data from the principal sources of mine acid.

3. In the deep mine and combined refuse pile-spoil bank model, overland as well as subsurface flows are determined by the Ohio State University Version of the Stanford Watershed Model. Stream flows and acid concentrations are calculated from the watershed model output.

4. The deep mine source model has been validated, but the refuse pile-spoil bank model has yet to be compared to suitable field data.

5. In general, sufficient field data is not now available to fully utilize the present models; additional hydrologic data is usually needed.

6. The optimization models provide a compact and efficient cost optimization algorithm capable of determining the least cost set of pollution control decisions for a branching array of acid sources.

7. The optimization model can also determine the distribution of pollution control decisions over the total array of acid sources which will produce the most desirable water quality for a fixed upper limit on dollars for pollution control.

8. The optimum resource allocation may vary with the "worst case" situation: (a) high flow, high acid load, or (b) low flow, high acid concentration.

SECTION II

RECOMMENDATIONS

The validity of the unit source models as related to "real world" situations should be established. The performance of the acid generation models should be evaluated by comparing predicted acid loads and in-stream concentrations with actual field data.

A sensitivity study of the key parameters in the models should be made; the degree of precision needed for these parameters, and methods for simplifying the amount and type of required input data need to be evaluated in relation to the degree of simulation success wanted.

Deficiencies in (normally) available field data should be described and guidelines for future acquisition and collection methodology developed.

Prepare readily usable models for EPA personnel, including instructional material for potential users with both detailed guidelines, card decks, tapes, as well as short courses when desired.

A predictive model should be developed for analyzing unmined areas to predict level of pollution to be expected as a function of the type and scale of mining anticipated, and to provide a basis for selecting mining and abatement methods to minimize pollution and cost of abatement.

The Optimization Model should be applied to analyze an existing watershed to assess the model's utility, ease of application, and potential results.

The effect of a reservoir as a component part of a watershed should be included in the Optimization Model so it could be evaluated as an alternative abatement method.

SECTION III

INTRODUCTION

The purpose of the work reported herein was to develop a planning and management tool for use in determining cost-effective actions to eliminate or abate acid mine drainage from coal mining operations. Specifically, computer based modeling techniques have been developed for the description and/or prediction of acidity levels in the drainage from deep and surface mines, and from coal refuse piles. Using these source models, the acid load from a given acid source or grouping of related sources can be predicted for any desired time period, with geological conditions and rainfall variation accounted for in the models. Knowing the acid load from each source, as a function of time, alternative mine drainage treatment and at-source abatement techniques can be identified and their costs estimated, together with their effectiveness in terms of reduction of the unabated acid loads. Alternative cost and effectiveness estimates for the individual acid sources are then used as input to an optimization model, which can be used to calculate the minimum cost for a given level of abatement in the basin, or, alternatively, the maximum water quality attainable for a given cost.

The approach used in the development of the computerized models outlined above was dictated by the current level of knowledge concerning the formation of acid in mines and refuse piles, the transport of acid to the receiving streams via ground and surface water runoff, and the cost and effectiveness of possible abatement and treatment alternatives. Due to the extremely short-term and seasonal variability in mine drainage quality and quantity from a given source, which is closely associated with local hydrology and precipitation patterns, and due to the extreme differences in the behavior of different mine types with regard to acid drainage production, caused by widely differing geologies and physical characteristics of the disturbed areas, empirical techniques for the description and prediction of acid load and drainage quantity have met with little success. Just as modern hydrologic models have developed into relatively detailed simulations of the actual physical processes occurring during rainfall and runoff, so must realistic acid mine drainage models simulate in some detail the physical and chemical processes known to occur. Thus, the source models developed in this work are mathematical simulations of the chemical reactions and water movements occurring in the actual sources as contrasted with empirical relationships derived by the fitting of curves to water quality data collected in the field. The use of deterministic, predictive models to describe the acid output from sources in a basin for a selected annual or multi-year precipitation sequence leads to the capability of being able to identify "worst cases" for each source, in terms of acid concentration or acid discharge rate. These worst cases often correspond to the spring flush from a deep mine, or an intense summer thunder storm on a refuse pile. By selecting feasible treatment or at-source abatement techniques, designed on the basis of "worst case" acid loads and total flows, corresponding alternative costs for abating and controlling mine drainage pollution can be optimized across the basin by a suitable optimization technique. The nature of the input cost and acid load data generated by the above techniques makes a partial enumeration algorithm for solving nonlinear integer programming problems the most suitable optimization technique.

The project approach outlined above led to the advisability of dividing project activities into four major subject areas. These were the hydrologic sub-model, the acid generation sub-model, abatement and treatment cost and effectiveness data accumulation, and the resource allocation optimization model. Work in these four major areas overlapped to a high degree, and frequent meetings of all project personnel, together with periodic meetings with the Project Officer were utilized to maintain coordination of effort. In the following sections, the theory and structure of the overall model are presented on a step-by-step basis, covering each component of the model, in turn. Illustrative examples of each of the model components follows, together with critical evaluation of the technique, and comments on the future development of this approach.

SECTION IV

SHORT DISCUSSION OF THE PROJECT MODEL CONTENT AND OPERATION

This section presents an overview of the various components or subprograms of the total project model, in a low key technical format, to provide the reader with a general understanding of its content and operation. Detailed technical discussions are given and are referred to in the Appendices. Hopefully, this method of presentation will permit the reader to understand the overall picture of the report contents without unnecessarily, at this phase of the reading, being burdened with highly technical, mathematical, and computer language passages.

The total model consists of rather distinct sub-parts and, therefore, for readability, these components are first discussed separately and then the complete project model can be seen by their integration. The discussion at this point will be limited to the basic composition of the sub models. Section V presents illustrative examples of the application of the model.

Component discussions will trace the project model development from basic physical phenomena through its cost effectiveness stage to the final phase of optimization of resource allocations.

BASIC PHYSICAL PHENOMENA

There are two basic physical phenomena associated with acid mine drainage. The first is the determination of the amount and movement of water associated with the various mine types; the second is the description of the rates of pollutant generation in these mines.

Hydrologic Model

The quantities of minewater flow or drainage are closely related to the concept of the hydrologic cycle. Hence, by modeling the hydrology of a watershed, the water activities in the various mine types can readily be accounted for by observing the respective portions of the hydrologic cycle model that apply to the waters involved in the mines. Discussions herein will now be limited, more or less, to the actual hydrologic model used in this project.

Hydrologic Cycle and its Model -

Three zones of hydrologic events can be assigned to describe a hydrologic cycle. Figure 1 shows a schematic diagram of the cycle as used by the model.

> Upper Zone--This zone is above the soil surface. It is used to describe the activities at and after precipitation above the soil surface. The activities include interception, transpiration, evaporation, overland flow, surface detention and depression storage.
>
> Lower Zone--This zone is the soil between the water table and land surface. It is used to describe the activities of infiltration from the upper zone, percolation, interflow, and the degree of soil moisture saturation.
>
> Deep Lower Zone--This zone is the soil below the water table. It is used to determine groundwater flow to the stream, and to deep storage (or aquifer). Because moisture percolates through the cracks and crevices of the aquifer, a time delay occurs between moisture entering the aquifer and leaving the aquifer as mine-water flow.

The hydrologic cycle is a continuous process. Precipitation falls on the upper zone and some of it will enter into the lower zone and deep lower zone. The balance will enter the upper zone (interception plus depression storage). To complete the cycle, evaporation and transpiration occur from all three zones.

History of the Hydrologic Model Development -

The various components of the hydrologic cycle can be described by mathematical expressions which can then be integrated to produce a total hydrologic model. These models, programmed for the high-speed digital computer, simulate the hydrologic cycle behavior in a basin. Several such models have been developed over the past decade. A very versatile and successful one, the Ohio State University version of the Stanford Watershed Model (SWM), was used, with slight modifications, in this project to calculate the quantities of water generated at the mine sites. A brief review of the history of this particular model follows.

The basic Stanford Watershed Model (SWM) was developed by Professors Crawford and Linsley at Stanford University in the early sixties. Dr. James, while at the University of Kentucky, modified the model

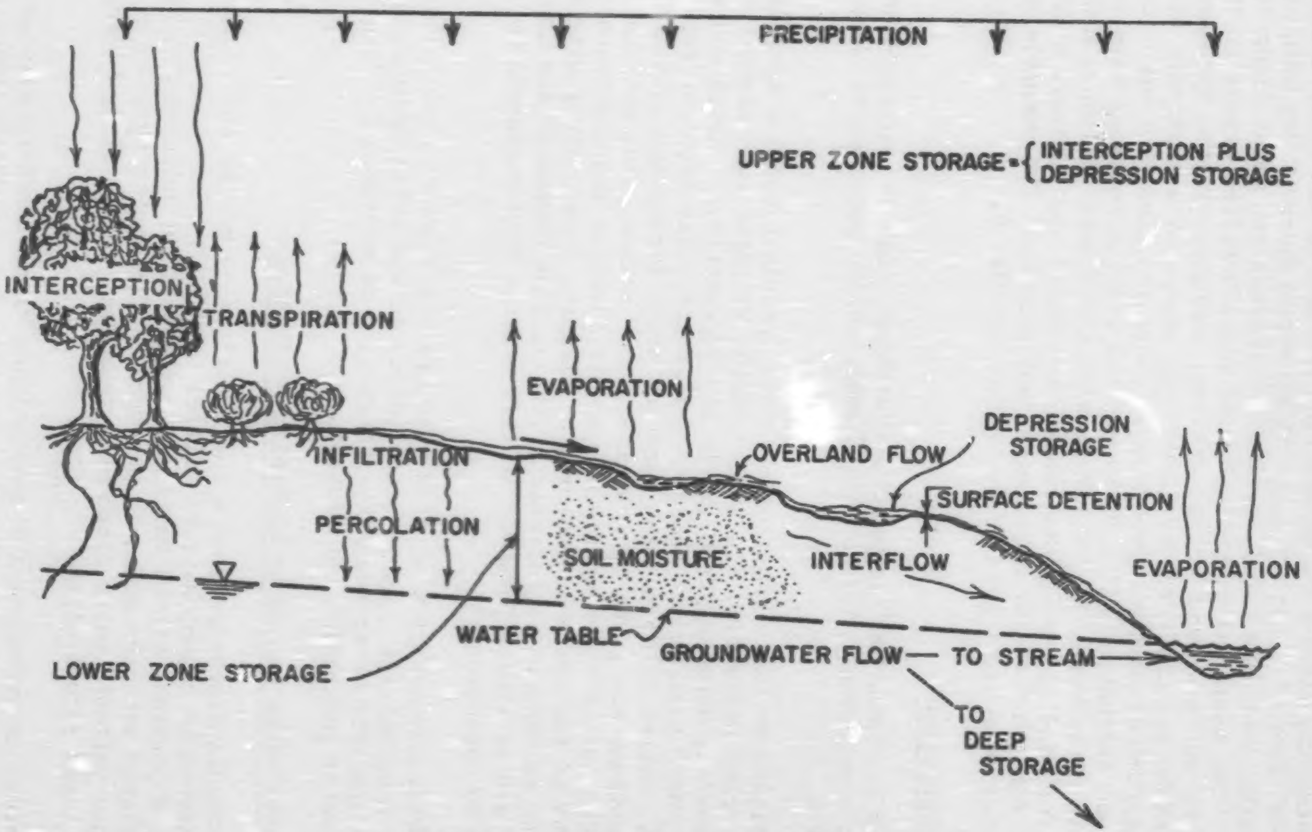

PRECIPITATION
UPPER ZONE STORAGE = { INTERCEPTION PLUS DEPRESSION STORAGE
INTERCEPTION
TRANSPIRATION
EVAPORATION
INFILTRATION
PERCOLATION
SOIL MOISTURE
OVERLAND FLOW
DEPRESSION STORAGE
SURFACE DETENTION
INTERFLOW
EVAPORATION
WATER TABLE
LOWER ZONE STORAGE
GROUNDWATER FLOW — TO STREAM
TO DEEP STORAGE

Figure 1. Schematic of hydrologic cycle

slightly and translated it into FORTRAN computer language in the mid-sixties. Various researchers at universities, governmental agencies, and consulting firms have worked with and proven the acceptabilities of the model over the years. Since 1968, researchers at The Ohio State University have made modifications to progress the model along these lines: the model was flow-diagrammed and a detailed exposé on the mechanics of its operation was written; computer plotted hydrograph and hyetograph programs were developed; key parameter sensitivity studies were performed; multiple groundwater recession phenomena were included; swamp and excessive soil shrinkage crack storage routines were created; snowmelt considerations for the Midwest areas were included; modifications for handling small watersheds were made; and finally, a User's Manual for the total modified model was written. The complete model and detailed evolution is presented in Appendix A.

Model Operation -

The basic scheme for the model's operation, that is, moisture accounting in a watershed, can be seen quickly in the logic block diagram of Figure 2.

In addition to the climatological data (precipitation, evaporation, wind, solar radiation, and temperature), 21 input parameters related to the physical aspects of the basin (12 measurable, 11 trial and adjustment, and 8 assigned or selected parameters), are required for the model. Tables in Appendix A describe these inputs and detailed explanations are given to determine their values. The mathematical formulations of the hydrologic concepts involved and the technical aspects of the computer programs to simulate these, along with their linking mechanisms represented by the various blocks in the logic diagram, are of little interest at this point. Full explanations are given in Appendix A.

Of particular interest at this time are the blocks in the diagram flagged by asterisks to indicate points in the total hydrologic model where specific water quantity information is accessed for individual mine water generation information.

Polluted Water Generated by Mining Activities -

The overall discharge from a basin containing mining activity can be considered as a composite of natural unpolluted flow emanating from the unmined portions of the basins plus the various polluted discharges associated with the mines in the basin. The mining activities could produce polluted water discharges from deep mines, strip mines, or associated refuse piles.

The modeling procedure for quantitizing these discharges is basically the same in all cases, that is, the use of a hydrologic model to

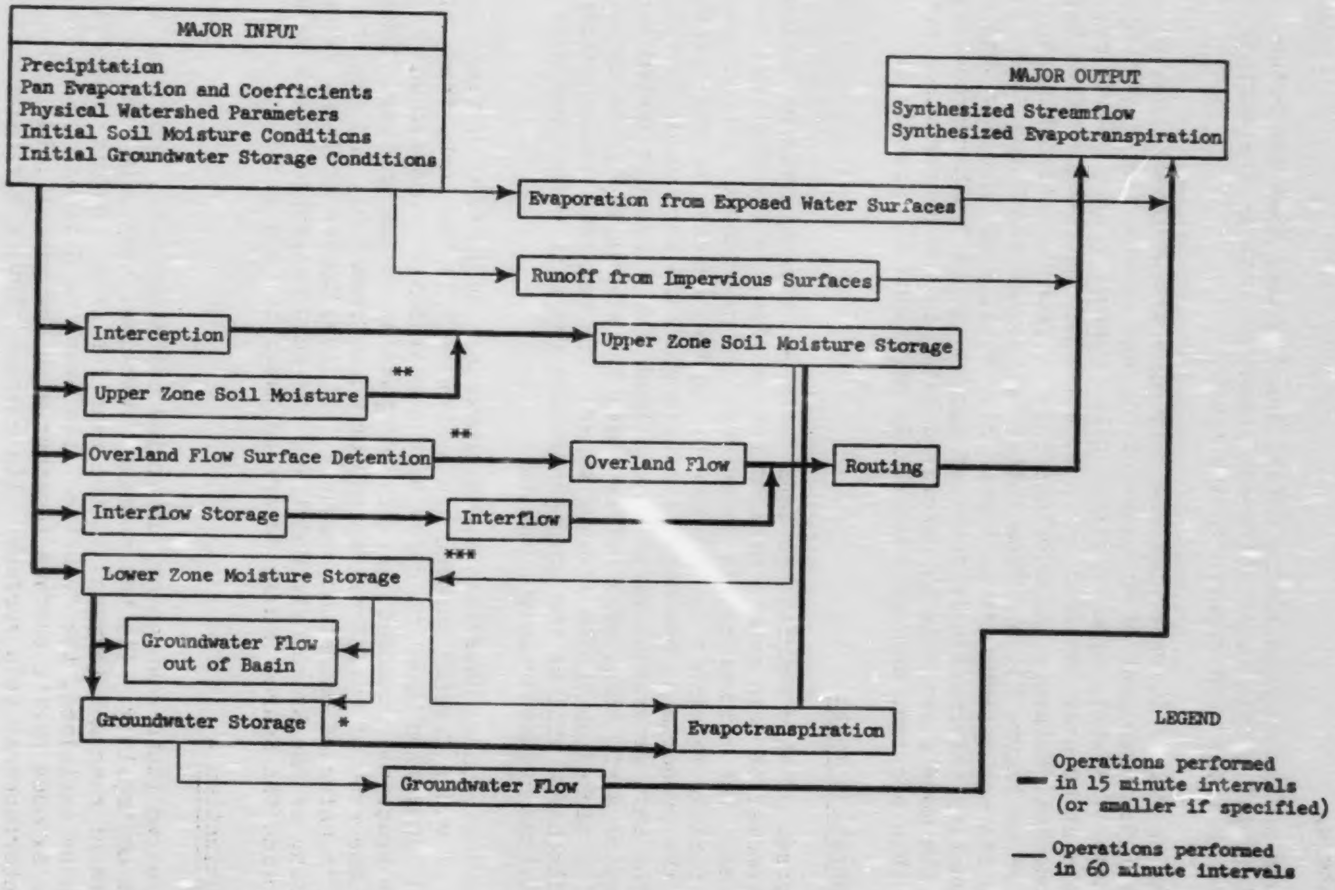

MAJOR INPUT
Precipitation
Pan Evaporation and Coefficients
Physical Watershed Parameters
Initial Soil Moisture Conditions
Initial Groundwater Storage Conditions
MAJOR OUTPUT
Synthesized Streamflow
Synthesized Evapotranspiration
Evaporation from Exposed Water Surfaces
Runoff from Impervious Surfaces
Interception
Upper Zone Soil Moisture Storage
Upper Zone Soil Moisture
**
Overland Flow Surface Detention
**
Overland Flow
Routing
Interflow Storage
Interflow
Lower Zone Moisture Storage
***
Groundwater Flow out of Basin
Groundwater Storage
*
Evapotranspiration
Groundwater Flow
LEGEND
Operations performed in 15 minute intervals (or smaller if specified)
Operations performed in 60 minute intervals

Figure 2. Moisture accounting in Stanford Watershed Model

predict the quantities of water involved and application of the appropriate pollution generation model to ascertain the quality of these waters.

The nature of the various mining activities dictates that the water generation will originate at different locations both in the earth's crust and in their counterparts in the model.

Referring to Figure 2, water for the deep mine situation is obtained from the model component of Groundwater Storage (point * on Figure 2). In the case of a strip mine or refuse pile, wherein acid products may accumulate at or near the soil surface, water may be obtained from the Upper Zone Soil Moisture and Overland Flow Surface Detention blocks (points ** on Figure 2), and from Interflow Storage and Lower Zone Moisture Storage (points *** on Figure 2). Details of just how this water quantity information is obtained and how it is further processed through the models are briefly described later in the source model portion of this section, and are technically explained in Appendix A.

Acid Generation Model

For purposes of this report, the term "acid generation" will be defined as the removal of acid from a mine or refuse pile via the drainage from the system. The acid materials so removed are provided by the oxidation of pyritic materials in the mine by oxygen, leading to the formation of the predominant products, sulfuric acid and ferrous or ferric iron. The acid generation rate or acid load from a pyritic system such as a deep or strip mine, or a refuse pile, is determined by the amount of acidity picked up by the flowing water. The net acid concentration in the discharge will be the acid generation rate, in weight per unit time, divided by the drainage flow rate.

The hydrologic model described above provides a definition of the amounts of water flowing through the system. It is the task, then, of the Acid Generation Model to describe two additional processes which are, for practical purposes, largely independent of one another. These are (1) the rate of pyrite oxidation (or acid formation), and (2) the rate of transfer of oxidation products (acidity) to the drainage water. The linking of the Hydrologic Model and the Acid Generation Model will then provide the Source Model, described later in this section.

Pyrite Oxidation -

There are two factors which may determine oxidation rate, depending on which is controlling: (1) the chemical reaction, and/or (2) rate of transport of reactant (oxygen) to the reaction site. Before this concept can be explained, the "reaction site" must be defined. Basically, it is an exposed pyrite surface together with the gas, liquid, and solid interfaces at this surface. Its characteristics are described by the interfacial area per unit volume of pyritic material and the conditions at this surface, particularly oxygen concentration. For practical purposes, bacterial catalysis of pyrite oxidation is insignificant, and, under conditions encountered in the field, the reaction can be considered first order with respect to oxygen (i.e., the rate varies in direct proportion to the oxygen concentration). A finite concentration of oxygen must be present at the pyrite surface before the surface can be termed a "reaction site".

A deep mine element, as shown in Figure 3 is a typical example of a pyritic system. Assuming the mined-out volume of a mine has an oxygen concentration of 21 percent, reaction sites will be exposed to oxygen concentrations varying from 21 percent at the working face to 0 percent back into the coal strata. The oxygen concentration profile and the net oxidation rate in a particular pyritic system depends on the void volume (porosity) and the exposed surface area of pyrite per unit volume. The calculation of the oxygen concentration profile is simply a problem of diffusion plus chemical reaction for which quantitative mathematical solutions are available. Specific examples are given in Appendix B.

Note that "reaction sites" extend as far into the porous media as oxygen diffuses. The greater the void volume, or porosity, of a pyrite-containing stratum, the greater the rate of oxygen diffusion because of the larger gas flow cross-sectional area available for diffusion. At the same time, more pyrite surfaces are exposed in a porous material simply because of the larger void space available for diffusion and because of the greater total surface exposed to the vapor phase. Conversely, the tighter the formation, the lower the quantity of oxygen which will diffuse through it, and the less oxygen which is available for oxidation per unit volume of material.

Since oxygen diffusivity in water is 1×10^{-4} that in air, essentially all oxygen must be transported to the reaction site as a vapor. Diffusion through water is insignificant, and the quantity of dissolved oxygen in water entering an underground mine is too small in itself to produce a significant acid load. This leads to the important fact that pyrite submerged under a pool of stagnant water, or enclosed in a porous medium which is saturated with water, will not be oxidized to any significant degree.

The general principles of Figure 3 still apply, in the case of a strip mine or refuse pile, except that oxygen diffusion would be from the soil-air interface down through the soil.

Another fact brought out by the conceptual model that must be kept in mind when interpreting discharge data is that the quality of effluent water is not directly related to the quality of water at reaction sites, that is, discharged water does not describe the aqueous environment at reaction sites in terms of concentration of oxygen, oxidation products, ferric/ferrous ratio, or other factors influencing the oxidation rate.

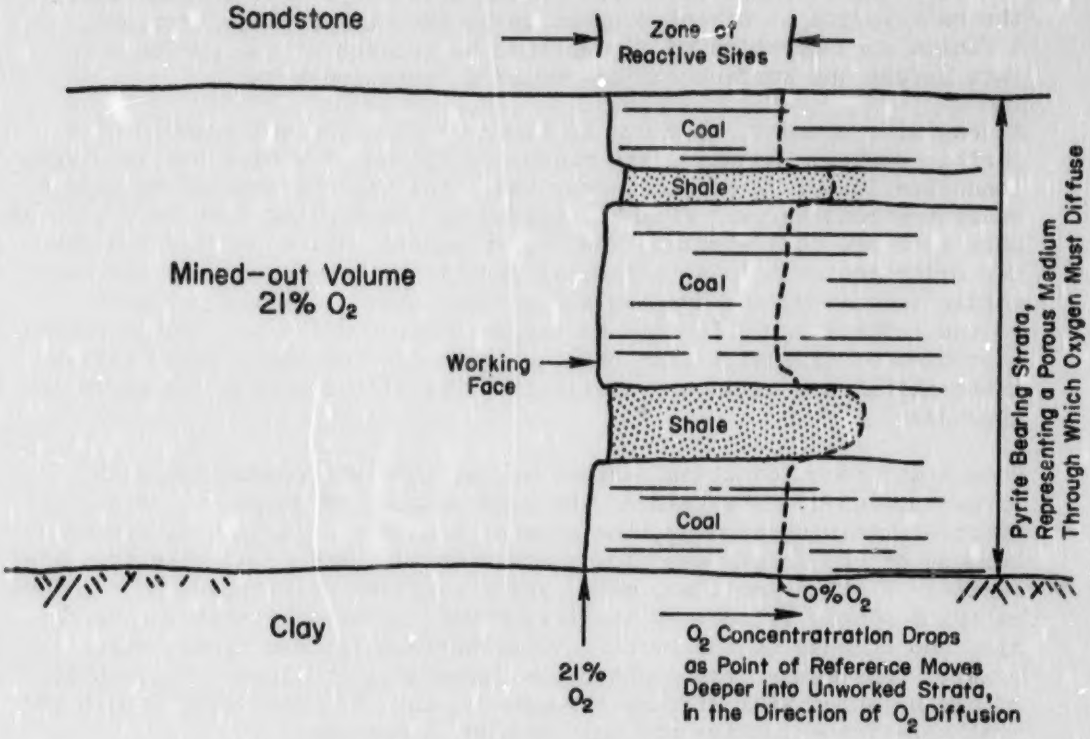

Sandstone
Zone of Reactive Sites
Coal
Shale
Mined-out Volume 21% O_2
Coal
Working Face
Shale
Coal
Pyrite Bearing Strata, Representing a Porous Medium Through Which Oxygen Must Diffuse
0% O_2
Clay
O_2 Concentratration Drops as Point of Reference Moves Deeper into Unworked Strata, in the Direction of O_2 Diffusion
21% O_2

Figure 3. Idealized deep mine element

Removal of Oxidation Products -

For purposes of discussion, the case of a deep mine will again be considered. In this case, the rate at which pyrite oxidation products enter the effluent stream is determined by three basic mechanisms:

1. Flushing by a rising groundwater table.
2. Percolation by water flowing down through porous zones and open channels or fractures during or after periods of heavy precipitation.

3. Diffusion or "weeping" of saturated solutions of reaction products caused by water condensing on reaction sites due to the lowered vapor pressure of the highly concentrated solutions at these locations.

The particular removal mechanism (or mechanisms) involved at a specific reaction site depends on its location in respect to the water table and/or flow channels through which ground water percolates. If a reaction site is isolated from these sources of direct removal, oxidation products will build up until the degree of saturation at the site and surrounding area is high enough that the rate of transport to points of direct removal by percolation or flushing is equal to the oxidation rate. Note that the build-up of oxidation products has no effect on the oxidation rate.

Obviously, the instantaneous rate of oxidation product removal, or acid generation rates, is a complex resultant of both pyrite oxidation rate and water movement in the system. Over a long term (measured, at the minimum, in years) the total amount of oxidation products removed may be equal to the total amount of oxidation product formed, but the daily or weekly acid loads, as measured from the discharge, cannot be related directly to the rate of pyrite oxidation.

All three of the above removal mechanisms have been observed either in the laboratory or in the mines. The flushing and percolation mechanisms are self-evident. The diffusion, or "weeping" process is too slow to be observed directly, but has been measured in the laboratory.

In the case of a refuse pile, pyritic material will generally be distributed throughout the pile, with only pyrite at and near the surface being exposed to oxygen. Spoil banks in a strip mine complex, on the other hand, may contain pyritic materials near the surface, or may be buried under a layer of nonpyritic, and hence, nonacid producing soil.

In either case, the same basic oxidation product removal processes can be active as in the case of the deep mine. However, due to the formation of acid at or near the surface, significant amounts of acid may appear in direct runoff and/or interflow, as well as in the base flow from the system. The direct runoff component is more of a "surface rinse" than a percolation or infiltration induced action, while the interflow is closely related to the percolation mechanism in a deep mine.

The differing physical structure of a deep mine, as contrasted with spoil banks and refuse piles, makes it impractical to develop a completely generalized Acid Generation Model applicable to both. Rather, separate models have been developed; namely, the Deep Mine Model, and the Combined Refuse Pile and Strip Mine Model. Both operate in accordance with the same principles in that they calculate both the rate of acid formation in the system and the release of acid to the system drainage in accordance with the predominant oxidation product removal mechanisms; both are linked with the Stanford Watershed Model to provide overall Source models. Block diagrams of the essential mechanisms of the two Acid Generation Models are shown in Figures 4 and 5. Detailed descriptions are given in Appendices A and B.

Pollutant Source Models

An acid source is defined, for purposes of this report, as a deep mine, strip mine, or refuse pile in which pyrite oxidation occurs. Depending on the scale of the system being simulated, a source might be a single mine or refuse pile, or a grouping of individual mines or piles having similar characteristics and treated, in aggregate, as a single source. All sources, regardless of type, have three basic characteristics which must be adequately simulated; the rate and physical location of pyrite oxidation in the source system, the transport of acid products from reactive sites to the effluent drainage from the source, and the dilution of this concentrated drainage by water in the receiving stream. As indicated previously, the net pyrite oxidation rate for a source is the solution to one or more sets of equations describing the diffusion of oxygen to the reactive sites and the consequent oxidation of the pyrite. The Acid Generation Models perform this calculation, giving the amount of acid produced in specific reaction zones. Transfer of this acid to the mine or pile drainage depends on the flow of water through or near the zones where the acid is formed, as reflected in the transport mechanisms. While it is computationally convenient to include in the Acid Generation Models the calculation of acid movement within the source and into the source drainage, these calculations require input from the Hydrologic Model to define the location and extent of water movement through the source. The Stanford Watershed Model, through its capacity to separately account for water storage and movement in the upper zone, lower zone, and deep lower zone, provides this input data. To the extent that water movement in the system will affect pyrite oxidation rates, this information must also be supplied by the Hydrologic Model. Lastly, the identifiable source drainage, together with its acid load, must be mixed with surface water flowing from non-acid portions of the basin under consideration, and a continuous accounting must be made of all water and acid in the basin. Thus, three requirements must be satisfied in the linking of the Hydrologic Model to the Acid Generation Models; the determination of acid movement and acid flow as a function of pyrite oxidation rate and a mass balance on both acid and water for the entire source-basin system.

Two source models have been constructed by linking the Hydrologic Model with the respective Acid Generation Models, yielding the Deep Mine Source Model, and the Combined Refuse Pile-Strip Mine Source Model.

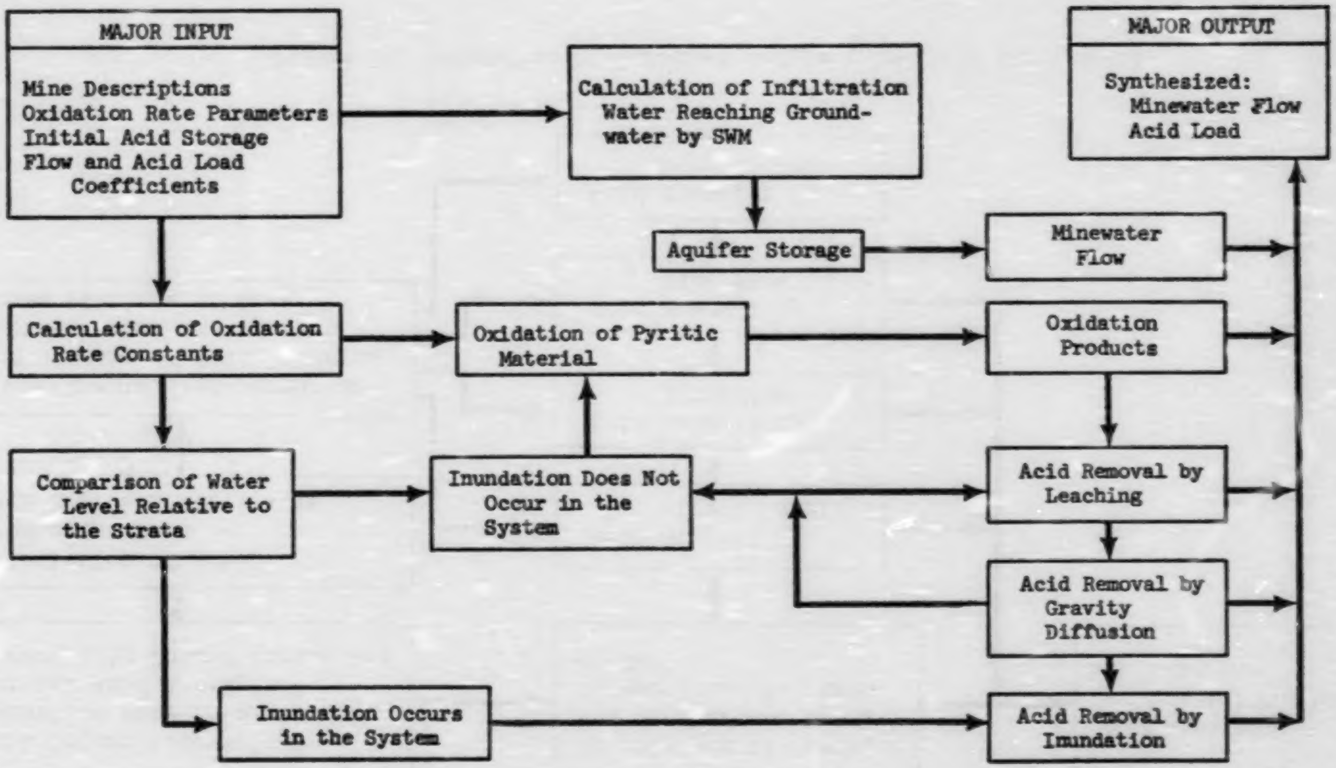

MAJOR INPUT
Mine Descriptions
Oxidation Rate Parameters
Initial Acid Storage
Flow and Acid Load Coefficients
Calculation of Infiltration Water Reaching Groundwater by SWM
MAJOR OUTPUT
Synthesized:
Minewater Flow
Acid Load
Aquifer Storage
Minewater Flow
Calculation of Oxidation Rate Constants
Oxidation of Pyritic Material
Oxidation Products
Comparison of Water Level Relative to the Strata
Inundation Does Not Occur in the System
Acid Removal by Leaching
Acid Removal by Gravity Diffusion
Inundation Occurs in the System
Acid Removal by Inundation

Figure 4. Schematic of Deep Mine Drainage Model

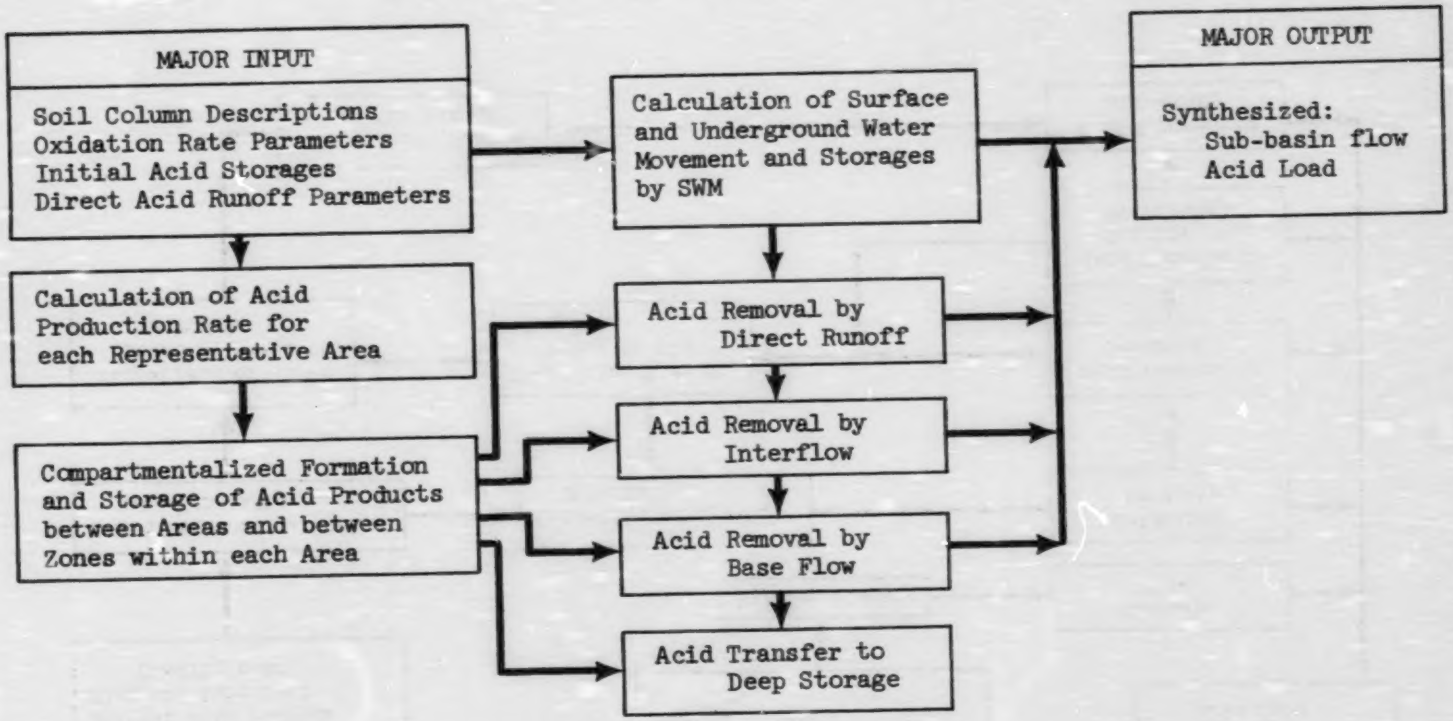

MAJOR INPUT
Soil Column Descriptions
Oxidation Rate Parameters
Initial Acid Storages
Direct Acid Runoff Parameters
Calculation of Acid Production Rate for each Representative Area
Compartmentalized Formation and Storage of Acid Products between Areas and between Zones within each Area
Calculation of Surface and Underground Water Movement and Storages by SWM
Acid Removal by Direct Runoff
Acid Removal by Interflow
Acid Removal by Base Flow
Acid Transfer to Deep Storage
MAJOR OUTPUT
Synthesized:
Sub-basin flow
Acid Load

Figure 5. Schematic of Combined Refuse Pile-Strip Mine Model

These models are presented through the use of block diagrams paralleling the actual sequence of the mathematical and computer processes (Figs. 6 and 7). Appendices A and B give detailed technical information on the model linking.

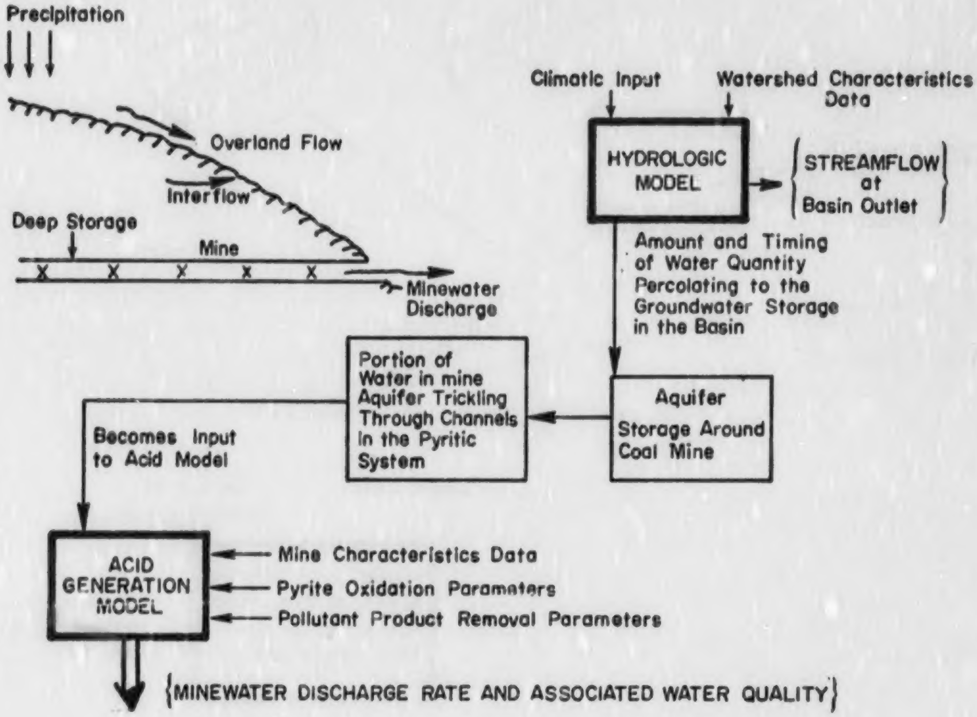
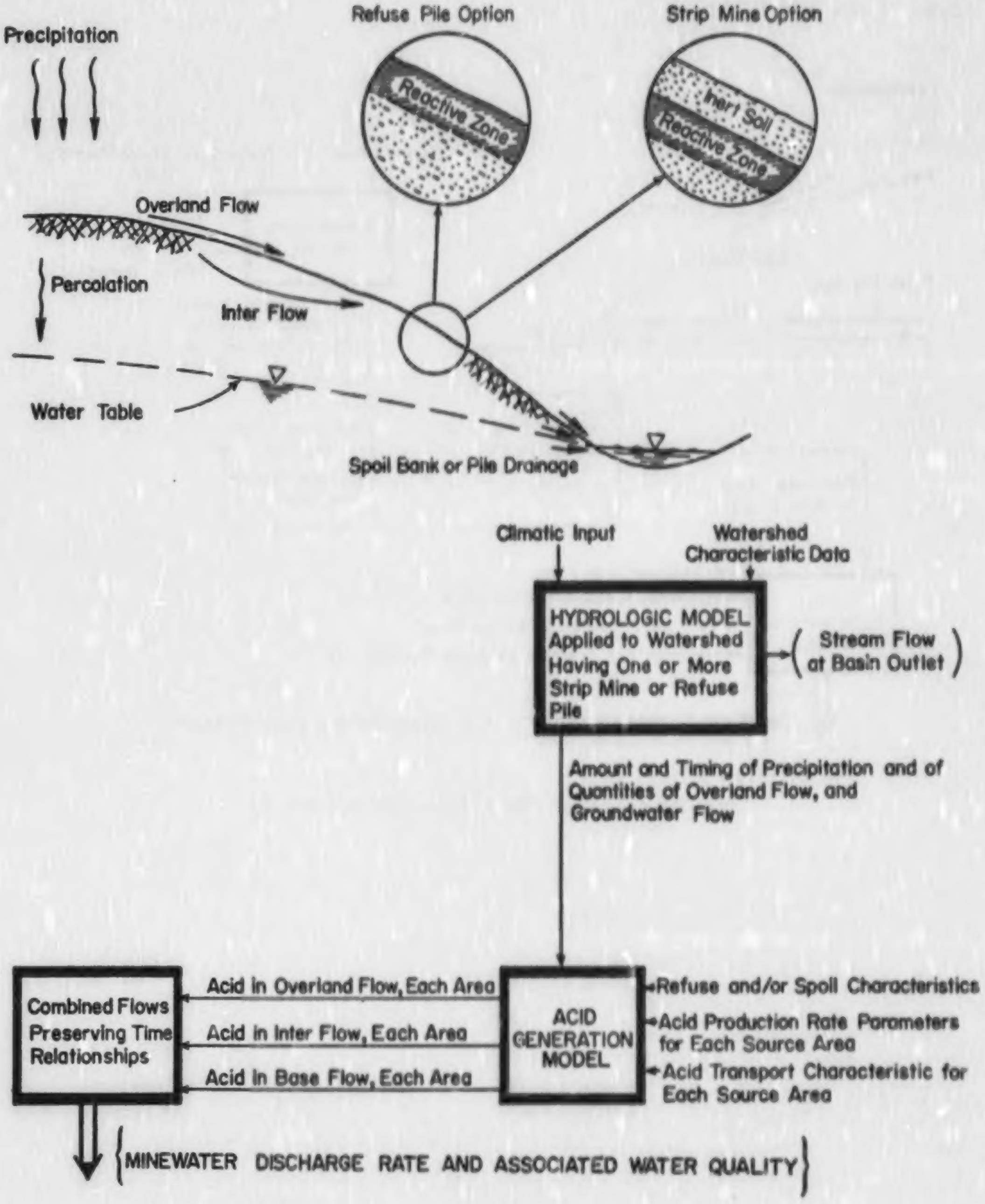

Precipitation
Overland Flow
Interflow
Deep Storage
Mine
Minewater Discharge
Climatic Input
Watershed Characteristics Data
HYDROLOGIC MODEL
STREAMFLOW at Basin Outlet
Amount and Timing of Water Quantity Percolating to the Groundwater Storage in the Basin
Aquifer
Storage Around Coal Mine
Portion of Water in mine Aquifer Trickling Through Channels in the Pyritic System
Becomes Input to Acid Model
ACID GENERATION MODEL
Mine Characteristics Data
Pyrite Oxidation Parameters
Pollutant Product Removal Parameters
{MINEWATER DISCHARGE RATE AND ASSOCIATED WATER QUALITY}

Figure 6. The Deep Mine Source Model



Precipitation
Refuse Pile Option
Reactive Zone
Strip Mine Option
Inert Soil
Reactive Zone
Overland Flow
Percolation
Inter Flow
Water Table
Spoil Bank or Pile Drainage
Climatic Input
Watershed Characteristic Data
HYDROLOGIC MODEL Applied to Watershed Having One or More Strip Mine or Refuse Pile
Stream Flow at Basin Outlet
Amount and Timing of Precipitation and of Quantities of Overland Flow, and Groundwater Flow
Combined Flows Preserving Time Relationships
Acid in Overland Flow, Each Area
Acid in Inter Flow, Each Area
Acid in Base Flow, Each Area
ACID GENERATION MODEL
Refuse and/or Spoil Characteristics
Acid Production Rate Parameters for Each Source Area
Acid Transport Characteristic for Each Source Area
MINEWATER DISCHARGE RATE AND ASSOCIATED WATER QUALITY

Figure 7. Total acid load and acid load from each source area

Figure 8 is a schematic diagram of a basin subdivided into three sub-basins, two of which have acid sources.

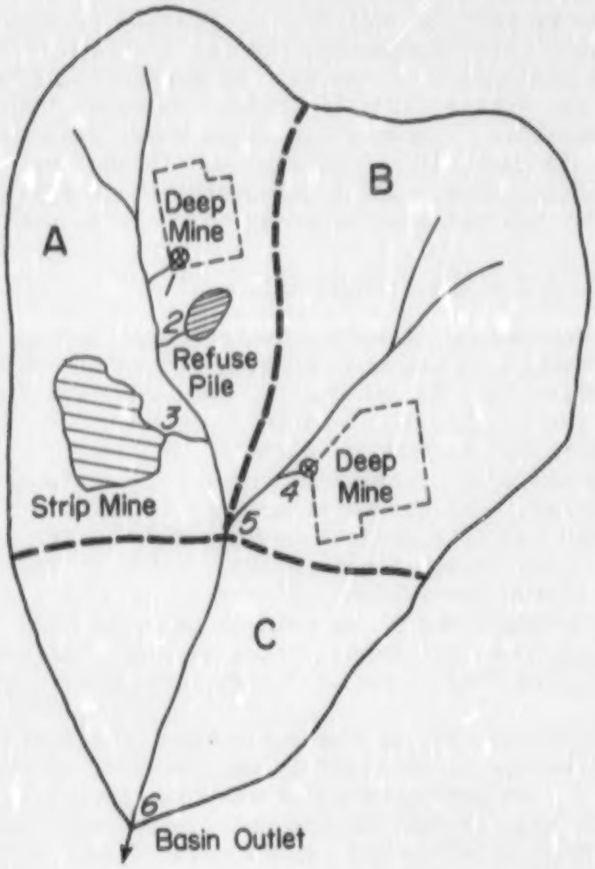


Figure 8. Schematic diagram of subdivided basin

Sub-basin A includes a deep mine, strip mine, and refuse pile, while Sub-basin B has only a deep mine. In the application of the Source Models to this example, both the Deep Mine Source Model and the Combined Refuse Pile-Strip Mine Source Model would be applied to Sub-basin A, with the latter model including both the strip mine and refuse pile in the same application. The Hydrologic Model would be applied only once over Sub-basin A, as it is common to both Source Models. The dual Source Model application would give the acid source drainage flow and acid loading rates at points 1, 2, and 3, and the flow rate at the basin outlet, point 5, all as a function of time for any time period (e.g., one year) and precipitation sequence desired. The source models identify the drainage flow rates and acid loading rates at the individual sources, but identifies the sub-basin flow only at the sub-basin outlet. Further, flow times in the stream above the sub-basin outlet are not calculated by the model, and water quantity upstream from point 5 can not be directly computed. In most applications, the simplifying approximation that all acid sources discharge into the stream at the sub-basin mouth will yield satisfactory results. If required, stream routing techniques could be applied within the sub-basin, but this is now considered to be an unwarranted refinement, and has not been built into the model.

The application of the Deep Mine Source Model to Sub-basin B would yield the mine drainage flow and acid load rates at point 4, and the sub-basin outflow at point 5. Again, the mine acid load would normally be assumed to discharge into the stream at point 5 for purposes of calculating the acid concentration at point 5 as a function of time.

Having estimated the flow and concentration at point 5 from both Sub-basins A and B, the flows can be added, normally with the assumption of complete mixing, to calculate an average acidity concentration. Between points 5 and 6, there are no acid sources tributary to the stream, and acid from above point 5 is diluted further by flow from Sub-basin C, and may be neutralized in some degree by alkalinity present in this flow. Stream routing times from point 5 to point 6 may or may not be taken into account, depending on the degree of accuracy required in estimating acid concentrations at point 6. The most appropriate method of combining sub-basin flows into the basin model varies with the individual case, and has not been included in the computer programs presented here. While decisions as to the degree of refinement required in accumulating sub-basin flows and acid loads depends heavily on field data available and judgment of the analyst, it is anticipated that stream routing requirements can be held to a minimum, particularly in the application of the "worst case" optimization procedure.

COST-EFFECTIVENESS MODEL

It is appropriate at this point to recapitulate the overall structure of the acid mine drainage abatement resource allocation procedure which has been developed on this project, to put cost and effectiveness determinations in the proper perspective.

The first step in the sequence is to define the nature and extent of the problem, which requires an estimate of the acid load from each source, as a function of time. This step, together with an estimate of the stream flow throughout the basin, provides a basis for the calculation of acid concentrations throughout the basin, again as a function of time. The Source Models described are designed to supply the acid loading information from each source and to provide the individual characteristics of each source.

The second step is the selection of alternative treatment and/or at-source abatement procedures which may be feasibly applied at the individual sources. The associated costs of each control alternative at each source, together with the effectiveness of each alternative in terms of acid reduction, becomes the basic input data for the selection of the optimum mix of treatment or at-source abatement efforts at each source. Further, the application of a uniform quality standard over the entire basin may be unrealistic, since existing or proposed land and water use may vary widely from point to point in the basin. Thus, some mechanism is necessary for varying the required water quality, or for weighting the value of good quality at different points throughout the basin. Considerations concerning these questions of cost, effectiveness, and valuation are discussed.

The third step is to determine the optimum distribution of resources to the control of acid drainage within the basin, using input from both Source Models, which define the pre-abatement condition of the basin, and the cost-effectiveness estimates for the various acid control alternatives.

Acid Control Alternatives

There are basically two approaches to acid mine drainage control, at-source abatement, and chemical or physical treatment of the drainage. Examples of at-source abatement include the sealing and flooding of deep mines, the inert gas blanketing of deep mines, and the regrading and covering of spoil banks and refuse piles, with or without lime or limestone application to the soil. All at-source abatement procedures have the common aim of eliminating, by some means, the exposure of pyritic materials to atmospheric oxygen. The primary advantage of at-source abatement is that, once accomplished, maintenance costs are low, or are eliminated altogether. The primary disadvantage is that the effectiveness of at-source abatement procedures in terms of acid load reductions are not very spectacular, or require long periods of time to reach maximum apparent effectiveness, due to the slow bleed-out of acid accumulated in the system prior to abatement activities.

Chemical or physical-chemical treatment, as opposed to at-source abatement, has the advantage of being quite positive in its effect. All of the acid can be neutralized and the soluble metals removed by appropriate chemical neutralization and precipitation. Even the anions, particularly sulfate, can be removed by such processes as reverse osmosis. However, operating costs can be quite high, sludge or brine disposal problems may be extremely difficult and expensive, and, the most sobering prospect of all, the required period of treatment may be, for all practical purposes, interminable. The "natural" burn-out of pyrite in mines, refuse piles, and spoil banks is measured not in years, but in decades or centuries.

The problem of estimating the cost and effectiveness of at-source abatement techniques is a difficult one, and procedures cannot be generalized. Field demonstration project results available at this time are disappointing, primarily because pre- and post-abatement drainage monitoring efforts have been generally inadequate for a definitive evaluation of the techniques applied. While the cost of sealing a mine or covering a refuse pile can be estimated by conventional procedures, the anticipated effectiveness of a given technique is difficult to predict. The effectiveness values estimated and reported for existing cases can not, in general, be taken at face value, and the judgment and experience of the analyst is an important factor in evaluating existing data.

As an alternative to the prediction of the effectiveness of proposed at-source abatement procedures on the basis of existing data, the use of the Source Models as predictive models for the evaluation of at-source abatement alternatives is strongly encouraged. These models are designed to respond to changes in oxygen availability to the pyrite in the system, and to changes in the hydrologic regimes affecting the system, all of which can be estimated for a given at-source abatement technique.

The costs and effectiveness of chemical and physical-chemical treatment alternatives can be closely estimated by conventional techniques once the drainage flow and acid loading characteristics of a given source are known. A review of reported costs of treatment by alternative methods was made, and is included in Appendix D. In general, cost will be a function of flow and loading, with unit costs decreasing with increasing scale of the treatment system. Details of the formulation of such functions are also given in Appendix D. Plant size is determined largely by the flow rate of the drainage to be treated, while the cost of treatment chemicals required is determined by the average annual acid load to the plant; both flows and loadings are provided by the Source Models. Identification of the capital cost of the treatment system, together with the estimated annual maintenance and operation costs, and appropriate amortization rates, provides the basis for calculation of an effective annual cost estimate for use in the optimization model. Here again, the wide variety of treatment alternatives available and the importance of taking the characteristics of the individual source into account in selecting feasible alternatives make a generalized computer program for this cost estimation step impractical.

Socio-Political-Economic Considerations

The decision to implement pollution control measures at mine sources in a watershed is influenced by socio-economic and political considerations. The importance associated with maintaining stream quality varies throughout a watershed. In particular, the land use in the vicinity of the stream should influence the requirement and importance of achieving desirable stream quality levels. Accordintly, the basin effectiveness measure defined in the following section provides for the influence of socio-political-economic considerations on the relative importance of individual stream reaches.

BASIN OPTIMIZATION MODELS

The Source Models can predict the acid load emitted from a particular site as a function of time; moreover, the acid load can be predicted for situations resulting from the application of pollution control measures taken to improve environmental quality. In addition, the Source Models can predict stream flows, and cost estimates can be generated for chemical treatment control measures at each pollution source. The Basin Optimization Models take these pollution loadings, control measures, and costs to determine optimal allocations of pollution control effort at each source with respect to the following criteria: (1) least cost allocation to achieve a specified quality level, and (2) most effective allocation for a specified cost. Relationships among these models are shown in Figure 9. The structure and use of the optimization models developed during this study follows.

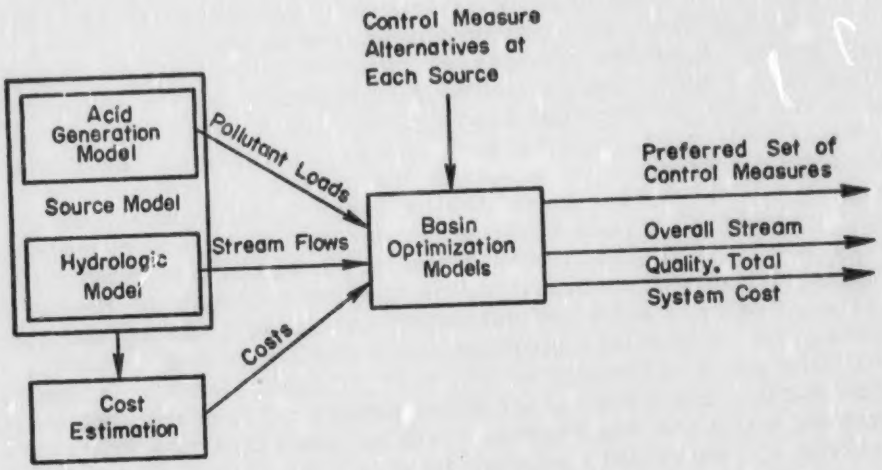

Control Measure Alternatives at Each Source
Acid Generation Model
Source Model
Hydrologic Model
Pollutant Loads
Stream Flows
Costs
Cost Estimation
Basin Optimization Models
Preferred Set of Control Measures
Overall Stream Quality, Total System Cost

Figure 9. Input-output diagram for Basin Optimization Models

The Basin Optimization Models represent the basin as a network of streams and a set of individual pollution or mine sources feeding these streams. Deep mines, strip mines, or refuse piles are all referred to as mine sources in the optimization model. The effluent from a mine source is assumed to enter one of these streams at a single point on the stream called a node. A single basin outlet stream is assumed to exist, and this basin outlet stream may have tributaries, and these tributaries may have tributaries. This network of streams creates a hierarchy among the streams which has, at most, a level of three. That is, the basin outlet is a third-level stream being fed by second-level streams, and the second-level streams are fed by first-level streams. A typical stream network is shown in Figure 10.

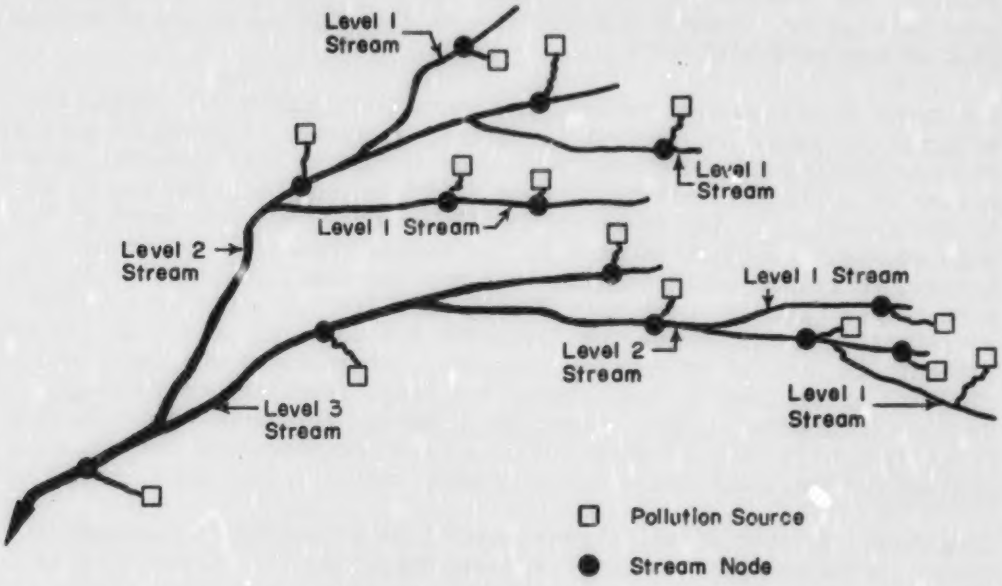

Level 1 Stream
Level 1 Stream
Level 1 Stream
Level 2 Stream
Level 1 Stream
Level 2 Stream
Level 3 Stream
Level 1 Stream
□ Pollution Source
● Stream Node

Figure 10. Stream network

Each mine source, noted as a small square in Figure 10, is represented as providing pollutant influent to the stream at a node point, and values for these pollutant flow rates in kilograms per hour are predicted by the Source Model as input data to the Basin Optimization Models. For example, the pollutant might be acid, and the pollutant flow rates would specify the total acidity of the effluent from each mine source. Also the stream carries a fluid flow, exclusive of pollutant, and the stream flow rates in cubic meters per second are predicted by the Unit Source Model as input data to the Basin Optimization Models at each node point. Values of these stream flow rates are assumed to be unaffected by actions to control stream pollution levels.

In addition to pollutant inputs from mine sources, the streams have natural pollutant inputs occurring throughout the network. These natural inputs are assumed to be distributed continuously between nodes, but the stream reach between each pair of nodes may have its own unique input rate. Thus, the natural pollutant input rate in kilograms per hour occurring between each pair of nodes is specified as input, and these natural pollutant inputs may have either positive or negative values. For example, a natural acid input rate of -0.05 kilograms per hour between two nodes would indicate an alkaline condition alleviating part of any potential acid mine drainage.

The nodes in the stream network are convenient points for specifying alternative pollution control decisions. A survey of possible methods for controlling mine drainage pollution indicated three general categories of pollution control methods as far as the optimization model is concerned: (1) Abatement at the mine source, (2) treatment at the mine source, and (3) treatment in the stream channel. Abatement is assumed to reduce pollutant flow but not necessarily eliminate it. A mine source produces pollutant flows, specified by input data to the optimization model, where the flow with abatement is less than the flow without the benefit of control measures. Examples of abatement are flooding or sealing of deep mines; covering, leveling, compacting, burying or grading of gob piles; and grading, covering, or replanting of strip mines. All of these methods have the potential for reducing pollutant flows, but their effectiveness varies from site to site.

Treatment, whether in the stream channel or at a site, is assumed to reduce pollutant flow to zero (without affecting the stream flow exclusive of pollutant), but treatment will not affect a condition where the stream pollutant measure is already negative. Using our acid mine drainage example again, the treatment facility will neutralize an acid stream until it has no total acidity, but it will not affect an alkaline stream. The assumption is being made here that once the decision is made to install a treatment facility, the most economical solution is to remove all acid conditions but do not change alkaline conditions.

Treatment facilities may be either at a mine source or in the stream channel. Source or site treatment is an option which is assumed to be available at any source, and site treatment is regarded as being able to treat all pollutant effluent before it reaches the stream. Only certain nodes, designated by input data, are capable of being locations for instream treatment facilities. These facilities, if implemented, will treat all fluid passing through a node; moreover, an instream treatment facility would treat the effluent from its local mine source before its effluent enters the stream channel. A typical stream network with potential instream treatment sites is shown in Figure 11.

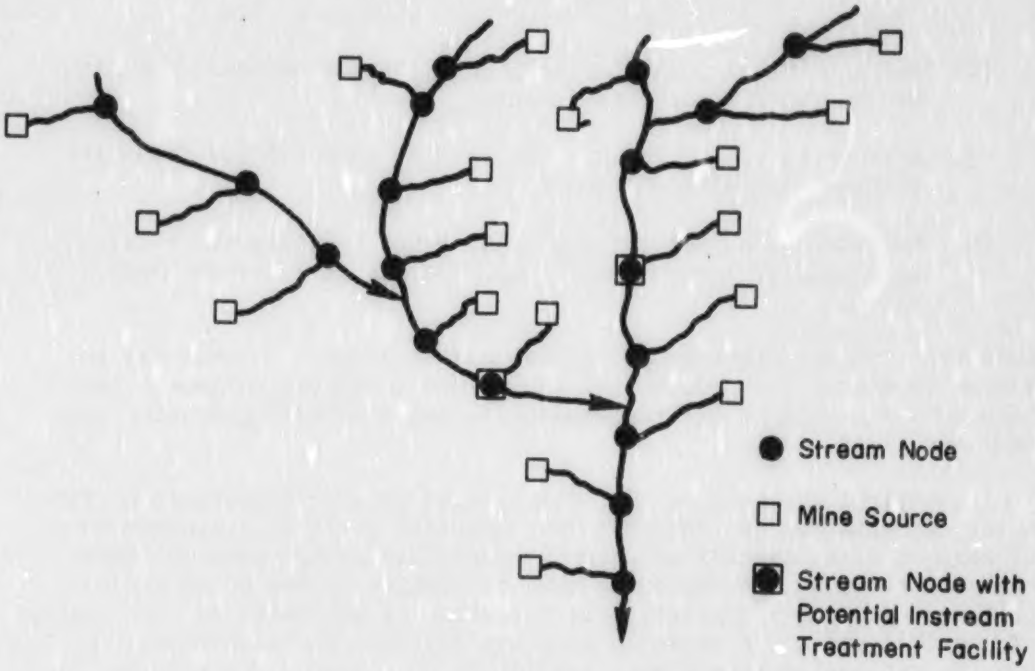


Figure 11. Illustration of stream network with potential instream treatment facilities

In addition to stream quality, pollution control costs are explicitly considered by the optimization algorithms. All costs incurred by these pollution control measures are assumed to be average annual costs which include recovery of capital, operating and maintenance costs. The cost to perform abatement at each site is specified by input data from the cost estimation step. Annual treatment costs involve two cost components, i.e., a variable cost that is directly proportional to the average annual amount of pollutant processed and the average annual cost exclusive of the variable cost component. An example of the variable cost component would be the cost of chemicals for neutralization of acid. The Cost Model provides as input data the variable cost to treat one unit of pollution in dollars per kilogram, and this value is assumed to be constant for all treatment processors. To calculate costs for treatment processors, the following cost data are specified by input for each mine or instream treatment site.

(1) Average annual cost of a treatment processor exclusive of variable cost for both instream processors and mine sources ($).

(2) Average annual pollutant load in kilograms emitted from mine sources without site abatement.

(3) Average annual pollutant load in kilograms emitted from mine sources with site abatement.

(4) Average annual natural pollutant input in kilograms between each pair of stream nodes (maybe either positive or negative).

Note that the optimization model must consider the fact that the instream treatment processors will experience an annual pollutant load that is a function of upstream abatement and treatment pollution control decisions.

A mathematical analysis of the optimization problem formulated to find a set of decisions at each node that constitute either a minimum cost of maximum effectiveness solution is described in Appendix C. This analysis revealed the following characteristics of the optimization problem: (a) the set of possible decisions is discrete, (b) the number of possible decisions increase very rapidly with the number of mine sources and instream treatment facilities considered; for example, two instream treatment facilities and fourteen mine sources imply over one billion uniquely different possible decisions, and (c) the constraint equations and criteria functions are nonlinear functions of the decision variables. For the above reasons, standard optimization methods such as linear programming and linear integer programming are inappropriate algorithms. Thus, an efficient nonlinear discrete optimization method had to be devised as a part of this research effort.

Another factor complicating any solution procedure is the dynamic stochastic nature of pollutant and stream flows; i.e., the pollutant flow rates and stream flows as predicted by the Pollutant Source and Hydrologic Models are time varying functions. Moreover, since these functions are strongly influenced by precipitation patterns, they are, in fact, stochastic. One way of handling their stochastic nature is to adopt a conservative approach by selecting very long time traces from the Source Models and analyzing these traces. Preferably, analyses of the Source Models can be conducted to indicate which precipitation patterns give the maximum pollution concentrations. Optimizing, using these precipitation patterns, will generate confidence that quality constraints will be maintained and that effective decisions will be generated. The basic assumption being made is that optimizing in this manner will yield a solution that provides quality and/or effectiveness measures at least as good as the worst case analyzed so frequently that any violations can be ignored.

Going one step further, a useful optimization algorithm can be obtained by completely suppressing the dynamic stochastic nature of the pollutant and stream flows. This approach will be called the "deterministic worst-case analysis." The basic idea inherent in this approach is to select a set of single values for pollutant and stream flows that represents the most adverse situation from a quality viewpoint. Then, an optimal solution using these values should almost always give better quality or more effectiveness in actual practice than considered in the solution procedure.

Two deterministic "worst case" Basin Optimization Models have been developed as a result of this research effort. They are defined below along with their identifying mnemonics:

1. Deterministic "worst-case" minimum cost (DWMC) model.

2. Deterministic "worst-case" maximum effectiveness (DWME) model.

The DWMC model takes a specified quality standard expressed in parts per million (ppm) that must be maintained throughout the stream network and determines a least-cost set of decisions achieving this standard. This set of decisions amounts to a specification of the treatment and abatement decisions at each mine site and treatment processor. There may be more than one set of decisions that give the same overall minimum cost; however, the DWMC model only provides one of these decisions as output.

The purpose of the maximum effectiveness (DWME) model is to allocate a fixed budget in the most effective manner. The maximum pollution control budget is specified as input data. The effectiveness measure also requires input data, but these inputs must be consistent with the measures used in the DWME.

The effectiveness measure has been designed to indicate the relationships between pollution levels, environmental impact, and land use in the vicinity of the watershed. These relationships are reflected in the effectiveness measure using two concepts:

1. The basic value of a stream based upon maximum pollution concentration level.

2. The relative importance of the stream between two nodes based upon its land use.

The above concepts are applied to individual stream reaches between adjacent stream nodes to compute a reach effectiveness measure which is the product of its relative importance and its basic value. Total watershed effectiveness is assumed to be the sum of the individual reach effectiveness measures. These concepts are described in more detail below and illustrated by examples.

The basic value of a stream reach is a number between zero and ten that indicates the reach's value based on observable effects from pollution concentrations. These effects include phenomena such as aquatic life, aesthetics, and water supply processing, which are assumed to be related to the maximum pollution concentration experienced. To portray the variation in basic value, maximum pollution concentrations are classified into intervals within which the observable effects are assumed to be constant. Table 1 illustrates the variation of basic value with pollution concentration.

Table 1. ILLUSTRATIVE VARIATION OF BASIC VALUE WITH MAXIMUM POLLUTION CONCENTRATION (ppm)

Maximum pollution concentration	Observable effects	Basic value
> 10	Fish cannot survive, noticeable odor, strong discoloration, water treatment costs increased by 100%.	0
8-10	Game fish cannot survive, high scavenger fish mortality, noticeable odor, water treatment costs increased by 50%.	2
6-8	High game fish mortality, scavenger fish will not reproduce, water treatment costs increased by 25%.	4
4-6	Game fish will not reproduce.	7.5
< 4	Aquatic life unaffected.	10

For each stream reach between two nodes, the basic value is determined and is weighted by the relative importance of the stream reach to give the effectiveness measure for this same stream reach. Relative importance is a quantity varying between zero and ten that specifies the importance of controlling pollution levels in each stream reach between adjacent nodes, land use in the vicinity of the stream reach, the impact of pollution on this land use, and the length of the reach to be considered in assigning relative importance values. The distribution of relative importance values throughout the watershed is determined in several steps. The first step is to select the most important stream reach (as defined by the stream between two adjacent nodes) which is given a relative importance value of ten. Then all other stream reaches are assigned values consistent with the difference between their importance and that of the most important stream reach.

The application of the above concepts is illustrated by the stream portrayed in Figure 12. The predominant land uses are noted in the figure. The most important reach, with respect to the impact of pollution, is the reach between nodes 1 and 2; thus, this reach is assigned a relative importance of 10.0. The other reaches are evaluated to have relative importances of 7.0 downstream of node 4, 5.0 between nodes 2 and 3, and 2.0 between nodes 3 and 4. Using these relative importances, the effectiveness measures for each stream reach can be obtained by determining basic values for each reach and multiplying these basic values by their respective relative importances. For specified abatement and treatment decisions, let the maximum pollution concentrations be 6.8, 5.9, 5.5, and 6.1 ppm, starting at the head of the stream (node 1) and proceding downstream. Using Table 1, the basic values for each reach are 4, 7.5, 7.5, and 4, respectively. Multiplying by the reach relative importances, the individual reach effectiveness measures are 40.0, 37.5, 15.0, and 28.0, respectively. Summing these reach effectiveness measures, the stream effectiveness measure is 120.5.

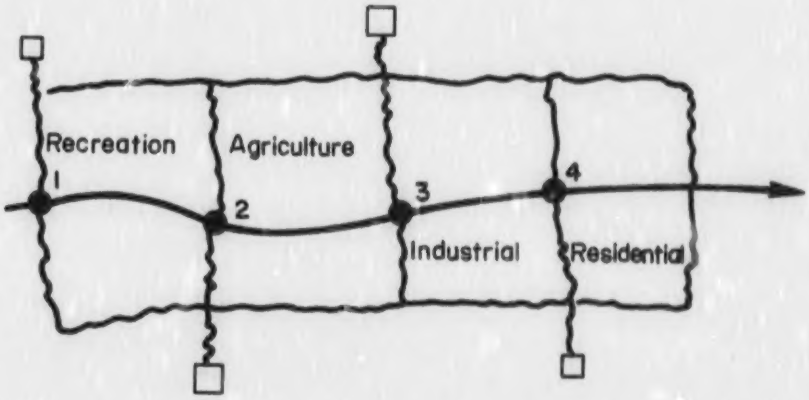


Figure 12. Single stream with adjoining land uses

Thus, to use the DWME model, a maximum annual cost budget, basic values, and relative importances must be specified as input data. It is possible, perhaps even likely, that there are many possible solutions that yield the maximum effectiveness solution. When additional solutions are encountered in the solution procedure giving maximum effectiveness within the budget constraint, the least-cost solution is recorded as optimal; however, the DWME algorithm was not explicitly designed to find the maximum effectiveness solution and then to determine the least costly way this solution was obtained. Thus, lower cost solutions for maximum effectiveness may exist. If the maximum effectiveness solution has the same basic value on all reaches, then the DWMC algorithm can be used to determine the least costly solution for maximum effectiveness.

Several other suggestions are offered to assist in the application of these models.

1. When a potential instream treatment site is not colocated with a mine source, then a dummy mine source can be created. This dummy source should have no pollutant effluent, and the cost to perform site abatement or site treatment should be a very large number.

2. Mines may exist that can not be controlled for technical reasons or because of legal or political constraints. When this case occurs, then the effluent from these mines may be regarded as a natural pollutant avoiding the use of nodes or decision variables.

SECTION V

ILLUSTRATIVE EXAMPLES

Since the basic composition and operation of the component project models have been discussed, we will show in this section, by sample applications, the operation of the models. The procedure will be to indicate what types of inputs are required and what typical results or outputs can be expected.

Critical discussions, on the rather limited applications to date, will be given later. Details on parameter value determinations, various types and format of data inputed, and the modeled results are given in the appendices.

For ease of reading, the two basic types of computer based sub-models developed on this project, the Source Models and the Optimization Model, are presented individually.

SOURCE MODELS

As seen earlier, the two types of source models, deep mine and combined refuse pile-strip mine, are basically of the same structure--a hydrologic model coupled to an acid generation model. Examples are given below of applications of both Source Models.

Finding suitable test sites poses great difficulty in that complete data on simultaneous hydrologic and acid minewater discharge are nonexistent or unknown to us. However, for a test of the Deep Mine Source Model, we were able to utilize a mine site where good mine drainage data were available, partial hydrologic data had been collected, and enough climatological data was available in the immediate region to reasonably assemble the remaining data. In a preliminary test of the Combined Refuse Pile-Strip Mine Source Model, field data were much less complete than in the case of the deep mine, and it was necessary to superimpose hypothetical strip mine and refuse pile areas on a real watershed for which hydrologic data were available.

Input Requirements

Given below is a descriptive list of the major information items required to operate the models:

Basin Information -

- Watershed drainage area
- Land use and distribution
- Flow capacity of main channel
- Mean overland flow path length
- Retardance coefficient for surface flows
- Average ground surface slopes
- Interflow and baseflow recession constants
- Channel routing parameters
- Index parameters reflecting interception, depression storage, infiltration, soil moisture storage, interflow movement, and groundwater movement

Climatic Data -

- Precipitation records
- Streamflow records
- Evaporation rates and coefficients
- Meteorological information for snowmelt

Deep Mine Information -

- Mine area
- Coal seam descriptors, materials, thickness
- Pyrite oxidation rate parameters reflecting diffusion, reaction, and temperature
- Acid transport parameters reflecting gravity diffusion, inundation, and leaching
- Initial acid storages
- Alkalinity conversion factors

Refuse Pile-Strip Mine Information -

- Strip mine and refuse pile areas
- Representative soil profiles of acid producing areas
- Pyrite oxidation rate parameters reflecting diffusion, reaction, and temperature
- Initial acid storages
- Acid transport mechanism parameters reflecting depth leached by direct runoff, leaching parameters, effective acid solubilities

Discharge Data -

Drainage flow records
Drainage quality records

Detailed descriptions of the information requirements, as applied to the examples shown, are given in Appendices A and B.

Examples of Output

Deep Mine Source Model -

A small drift mine (McDaniels) in Southern Ohio was chosen for a test application of the Deep Mine Source Model. Details of the history of this mine site and sources of data are given in Appendix A.

Simulation outputs are obtained both in tabular and graphical form. All of the following sample output is shown in expanded form in Appendix A. The figures (13,14,15) below are not actual ones but, rather, condensations to show what a typical plotted output of a computer run gives. The plots are labeled and what they represent is self evident.

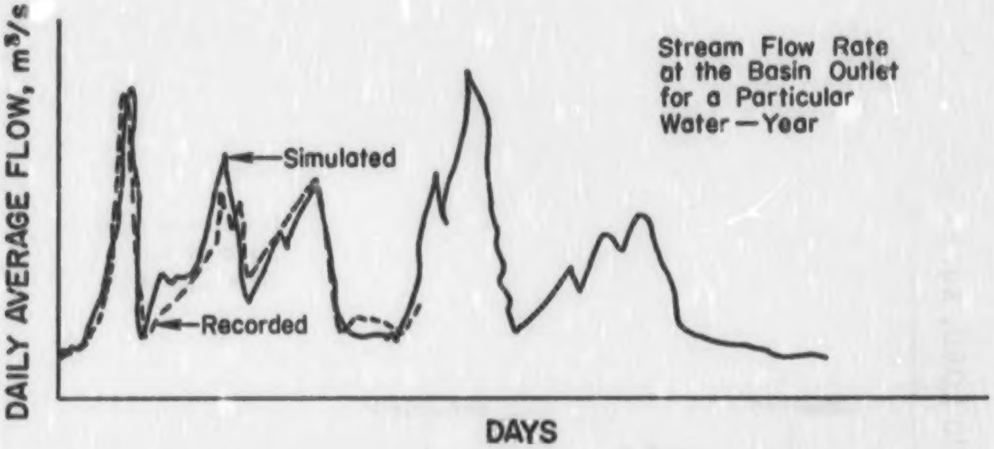


Figure 13. Streamflow hydrograph at the Big Four Watershed Outlet

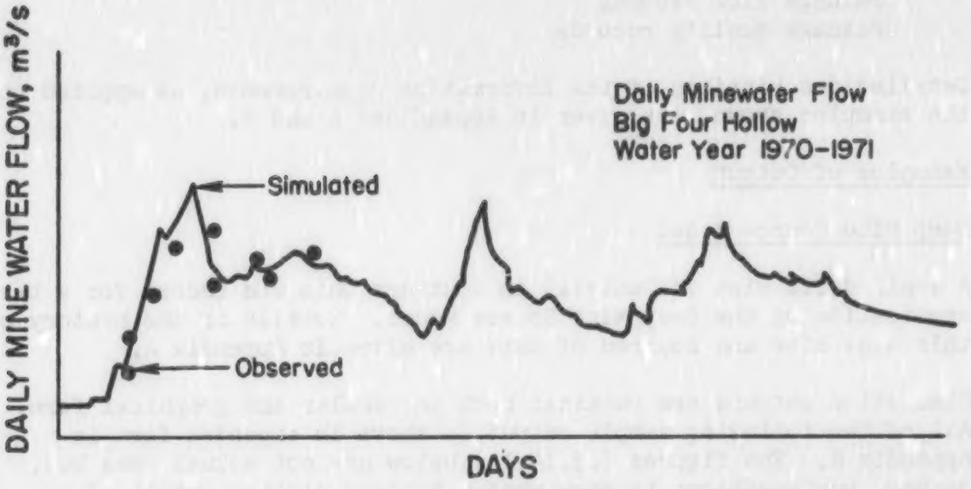


Figure 14. Daily minewater discharge from McDaniels' Mine

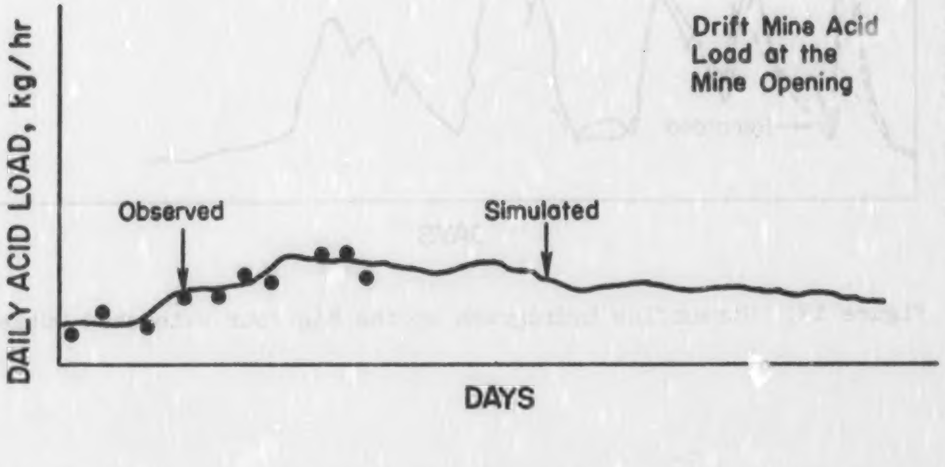


Figure 15. Daily acid load discharge from McDaniels' Mine

If the model is to be used to evaluate a proposed at-source abatement technique such as mine sealing, the specific deep mine input information is modified to reflect the effects of sealing, and the model is run as above. In this manner, both the short and long term effect of at-source abatement alternatives may be tested. The analyst so engaged is referred to Appendix A, and to the report by Morth et al. entitled, "Pyritic Systems: A Mathematical Model."

Combined Refuse Pile-Strip Mine Source Model =

Since an adequate test site for this Source Model was not available, hypothetical strip mine and refuse pile areas were superimposed on Watershed 94 of the North Appalachian Experimental Watershed near Coshocton, Ohio. The strip mine and refuse pile area of this watershed were assumed to be 10% and 2%, respectively. Details of the sources of data, site history, and assumed acid producing characteristics are given in Appendix B.

For this model, simulation outputs are obtained only in tabular form and may be plotted by machine. The figures (16 through 20) shown below are graphical representations of the output. The plots are labeled, and what they represent is self evident from their titles. Note that only simulated data are shown, since there was no recorded data for this hypothetical example.

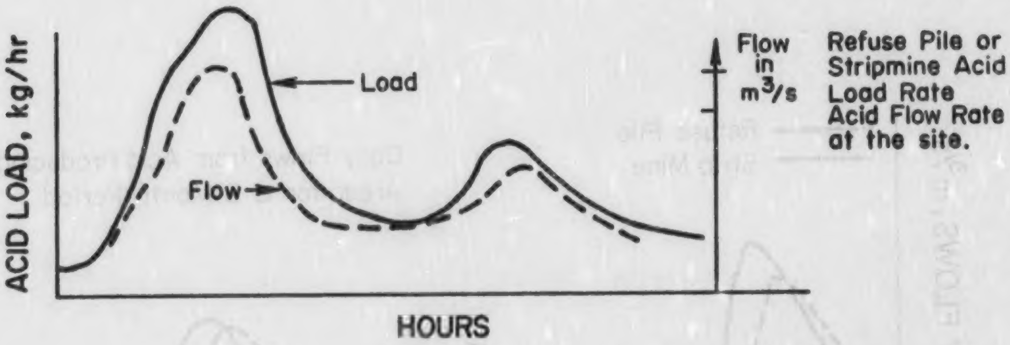


Figure 16. Acid load nonuniform short duration rain

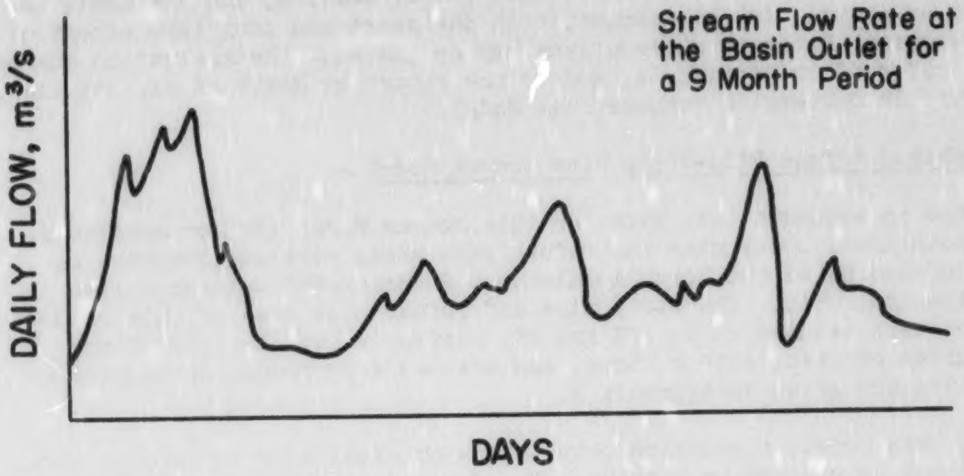


Figure 17. Stream flow hydrograph at the basin outlet

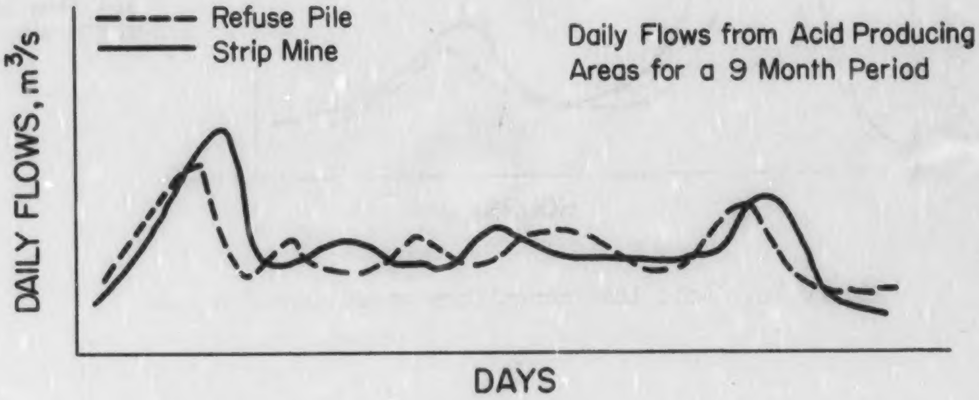


Figure 18. Daily flows from acid producing areas

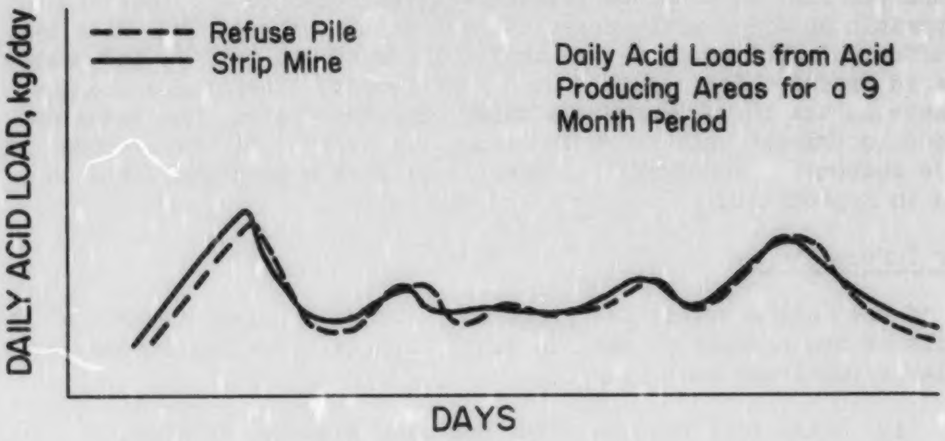


Figure 19. Daily acid loads from acid producing areas

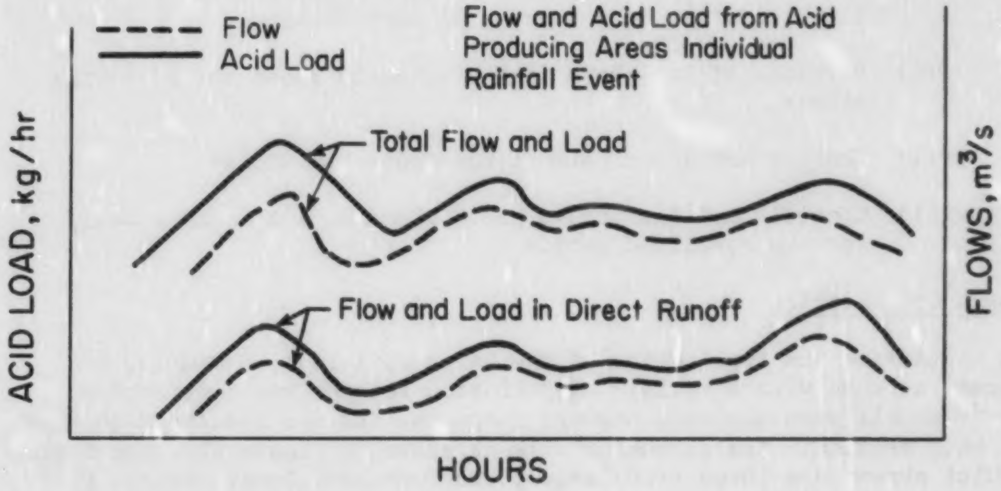


Figure 20. Flow and acid load from acid producing areas - short duration rain

As in the case of the Deep Mine Source Model, the Combined Refuse Pile-Strip Mine Source Model can be readily applied to the prediction of the effectiveness of at-source abatement techniques. For example, regrading and covering of a refuse pile effectively changes the location of the pyritic material with respect to the soil surface. This will have the effect of decreasing oxygen availability to the pyrite, and making the acid produced inaccessible by direct runoff. When these changes are made in the input parameters describing the system, the model can be used to predict both short and long term effects of the proposed at-source abatement procedure. An example of such a predictive run is shown in Appendix B.

Other Information -

Both of the Source Models are programmed to produce many aspects of the details of the process as well as final outputs. Tabular output is available for items such as:

(i) Daily infiltration water reaching the mine aquifer.

(ii) Average daily streamflow rates leaving the watershed.

(iii) Average daily flow rates from drift mine opening.

(iv) Average daily acid load (by origin component and total) issuing from drift mine.

(v) Daily acid load from a refuse pile direct runoff, interflow, baseflow, and then total contribution.

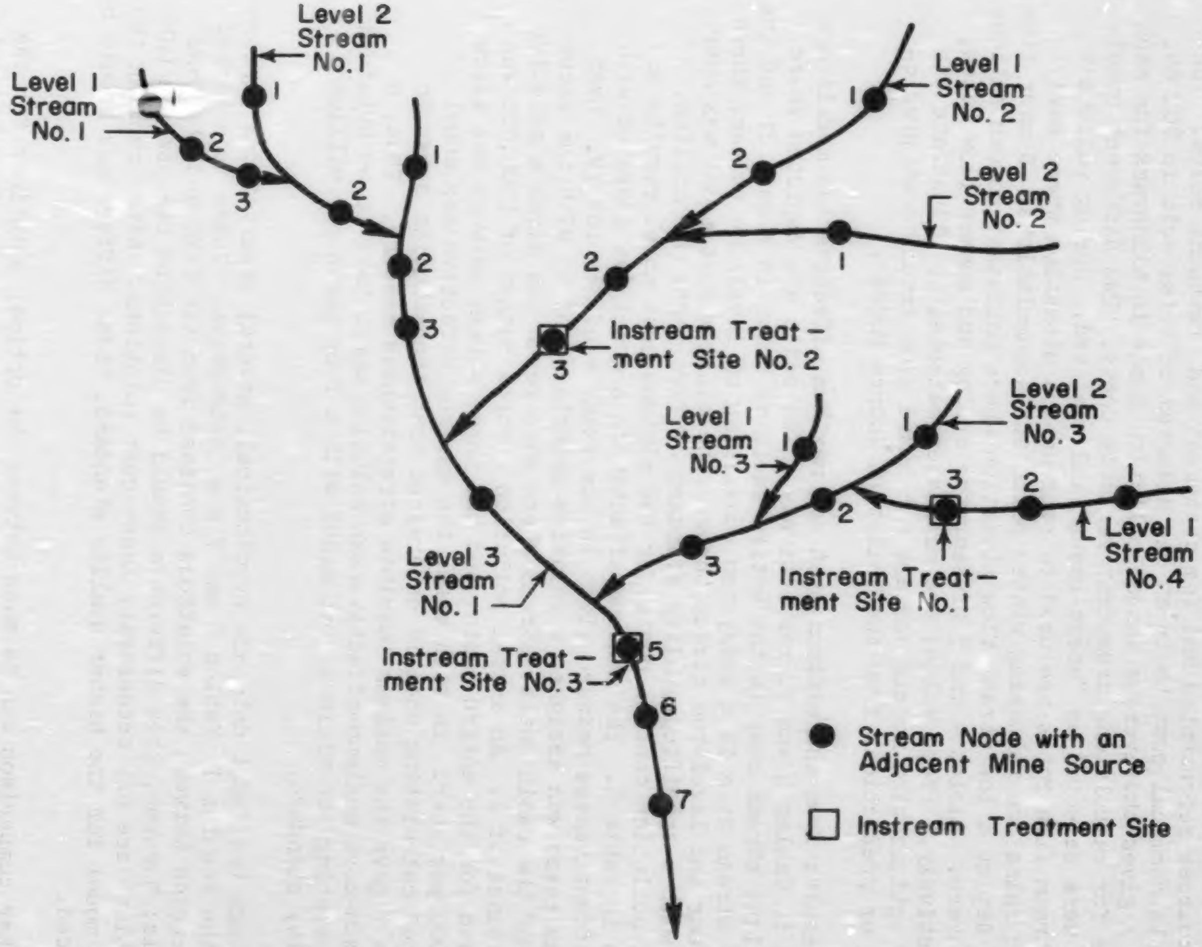
(vi) Portions of the above that eventually reach the receiving stream.

(vii) Monthly and annual sum of the above quantities.

(viii) Specific details of the above items on a 15-minute basis for any specified period

Basin Optimization

To illustrate the application of the basin optimization models, a stream network with a variety of pollution sources was constructed. These models were applied, answers computed, and the results described in this section. The stream network is shown in Figure 21. The basin outlet stream has three tributaries, and four additional streams feed these tributaries. Thus, there are four level one streams, three level two streams, and one level three stream. Note the three potential instream processor sites, and also observe that the third instream



Level 1 Stream No. 1
Level 2 Stream No. 1
Level 1 Stream No. 2
Level 2 Stream No. 2
Instream Treatment Site No. 2
Level 1 Stream No. 3
Level 2 Stream No. 3
Level 1 Stream No. 4
Instream Treatment Site No. 1
Level 3 Stream No. 1
Instream Treatment Site No. 3
1
2
3
5
6
7
● Stream Node with an Adjacent Mine Source
□ Instream Treatment Site

Figure 21. Stream network

processor site is not colocated with a mine source. Thus, a dummy mine source will be used at the fifth node on level three stream number 1.

Tables 2 through 5 present the input data for this basin required by the optimization models. Table 2 presents a short description of the mine sources represented and the abatement and treatment costs. The variable chemical cost to treat one kilogram of total acid is $0.264. Table 3 gives the average annual pollution loads in kilograms for each source for calculating treatment variable costs. Two different conditions were selected as "worst-cases" and analyzed. Spring pollutant and stream flow rates are used to represent a situation where early spring rains are releasing winter pollutant accumulations. Summer flow rates depict a low stream flow situation where pollutant concentrations are severe. Tables 4 and 5 present the spring and summer flow rates, respectively. These values are estimates selected to illustrate the basin optimization models and are not based upon actual cost predictions or predictions from the Pollutant Source Models.

The results from the minimum cost and maximum effectiveness models are shown in Tables 6 and 7, respectively. Two cases are depicted where the eight stream case is the entire basin as shown in Figure 21 and the three stream case is a subsystem consisting of level-one streams three and four and level-two stream three. The minimum cost model was operated with a specified quality standard of five parts per million. Total pollution control costs for the minimum cost model results are shown in Table 8. The maximum effectiveness model was operated with the effectiveness measure given in the example of Section IV. Each stream reach was assigned a relative importance of 10 with the exception of the basin outlet stream where each reach was given a relative importance of 1. An annual pollution control budget of $300,000 was allowed for the entire basin, and the three stream network was allowed $85,000 per year. In each case, the maximum effectiveness model stopped calculating when it determined that the solution shown in Table 7 gave the maximum possible effectiveness measure. Thus, a minimum-cost maximum-effectiveness solution would have to be obtained by operating the minimum cost model with a four parts per million quality standard.

Although the input data are hypothetical, several observations concerning the results in Tables 6 and 7 are instructive. There is a marked difference between the solutions obtained from the DWMC and the DWME models; however, this difference should be discounted because the DWME solutions are not necessarily least-cost solutions. After running the DWMC model for the higher quality standard, these differences should be reduced.

Another comparison can be made between the optimal solutions for the three stream subsystem and the complete eight-stream basin. The solutions are identical in all cases for the minimum cost model;

Table 2. MINE SOURCES AND POLLUTION CONTROL COSTS

Node	Description	Abatement cost ($1/yr)	Treatment processor cost ($1/yr)
Level 1 streams			
Number 1			
1	Small drift mine	3000	2500
2	Small gob pile	2250	2000
3	Small drift mine	3000	1800
Number 2			
1	Large underground mine	30000	15000
2	Large gob pile	15000	5000
Number 3			
1	Large drift mine	22500	11000
Number 4			
1	Large drift mine	30000	10000
2	Large gob pile	22500	6000
3	Small drift mine	3750	2500
Level 2 streams			
Number 1			
1	Large underground mine	15000	2200
2	Spoil banks	15000	2000
Number 2			
1	Large underground mine	30000	17000
2	Large gob pile	22500	4000
3	Spoil banks	15000	5000
Number 3			
1	Small drift mine	6000	4000
2	Large gob pile	30000	4000
3	Large underground mine	22500	12000
Level 3 streams			
Number 1			
1	Small drift mine	3000	3000
2	Spoil banks	7500	2500
3	Small drift mine	3750	7000
4	Small drift mine	4500	2000
5	Dummy source	9999999	9999999
6	Large gob pile	22500	2200
7	Large underground mine	22500	10000

Table 3. AVERAGE ANNUAL POLLUTANT LOADS (kg)

	Annual pollutant no abatement	Annual pollutant abatement	Annual natural pollutant
Level 1 streams			
Number 1			
1	2041	680	-181
2	2041	227	-181
3	2041	680	-181
Number 2			
1	163292	54431	-181
2	49895	4536	-227
Number 3			
1	30617	2268	+45
Number 4			
1	20412	9979	0
2	6804	3402	-45
3	3084		-45
Level 2 streams			
Number 1			
1	14968	1361	+45
2	6123	2268	+45
Number 2			
1	76657	36287	-91
2	113398	4536	-136
3	18144	1361	-181
Number 3			
1	4536	907	-227
2	113398	3629	-227
3	25401	11340	-181
Level 3 streams			
Number 1			
1	998	454	-91
2	3084	454	-91
3	3084	1361	-91
4	6123	2722	-91
5	0	0	0
6	49895	2268	-181
7	12247	4082	-272

Table 4. POLLUTANT AND STREAM FLOWS SPRING RATES

Node	Pollutant flow no abatement (kg/hr)	Pollutant flow abatement (kg/hr)	Natural pollutant flow (kg/hr)	Stream Flow (m^3/sec)
Level 1 streams				
Number 1				
1	0.91	0.27	-0.009	0.15
2	0.009	0.005	-0.009	0.18
3	1.00	0.32	-0.009	0.20
Number 2				
1	68.04	22.68	-0.009	0.30
2	0.91	0.45	-0.014	0.70
Number 3				
1	36.29	2.27	-0.005	0.18
Number 4				
1	20.41	9.07	-0.014	0.46
2	1.36	0.91	-0.014	0.70
3	1.81	0.91	-0.014	0.79
Level 2 streams				
Number 1				
1	18.14	6.80	-0.009	0.30
2	0.91	0.45	-0.009	0.82
Number 2				
1	45.36	22.68	-0.014	0.24
2	1.81	1.36	-0.014	1.52
3	2.27	0.45	-0.014	1.65
Number 3				
1	3.63	0.91	-0.018	0.37
2	0.91	0.68	-0.018	1.40
3	25.40	11.34	-0.018	2.07
Level 3 streams				
Number 1				
1	1.81	0.68	-0.009	4.57
2	0.36	0.09	-0.009	5.49
3	0.91	0.45	-0.009	5.52
4	4.54	2.27	-0.009	7.47
5	0	0	0	10.21
6	0.91	0.68	-0.014	10.21
7	18.14	6.80	-0.018	10.36

Table 5. POLLUTANT AND STREAM FLOWS SUMMER RATES

Node	Pollutant flow no abatement (kg/hr)	Pollutant flow abatement (kg/hr)	Natural pollutant flow (kg/hr)	Stream flow (m^3/sec)
Level 1 streams				
Number 1				
1	0.23	0.009	+0.005	0.02
2	2.04	0.227	+0.005	0.02
3	0.045	0.014	+0.005	0.06
Number 2				
1	0.91	0.32	+0.009	0.09
2	54.43	0.45	+0.009	0.12
Number 3				
1	0.544	0.023	+0.014	0.09
Number 4				
1	0.227	0.18	+0.005	0.21
2	68.04	1.81	+0.005	0.24
3	0.036	0.027	+0.005	0.46
Level 2 streams				
Number 1				
1	0.36	0.14	+0.005	0.06
2	0.36	0.14	+0.005	0.18
Number 2				
1	1.81	0.91	0	0.09
2	68.04	2.27	0	0.36
3	1.36	0.09	-0.005	0.55
Number 3				
1	0.023	0.004	-0.005	0.12
2	56.70	18.14	-0.005	0.76
3	0.27	0.14	-0.009	1.10
Level 3 streams				
Number 1				
1	0.023	0.009	0	1.83
2	0.18	0.045	0	2.59
3	0.023	0.009	0	2.71
4	0.068	0.036	0	3.66
5	0	0	0	5.33
6	68.04	1.81	-0.005	5.33
7	0.36	0.14	-0.009	5.52

Table 6. MINIMUM COST (DWMC) MODEL RESULTS

Table 7. MAXIMUM EFFECTIVENESS (DWME) MODEL RESULTS

	Table 6				Table 7			
	Spring rates		Summer rates		Spring rates		Summer rates	
Node	8 streams	3 streams	8 streams	3 streams	8 streams	3 streams	8 streams	3 streams
Level 1 streams								
Number 1								
1	Treat		Abate		Treat		Treat	
2	--[b]		Treat		--		Treat	
3	Treat		--		Treat		--	
Number 2								
1	Treat		Treat		Treat		Treat	
2	Abate		Treat		Treat		Treat	
Number 3								
1	Treat	Treat	Treat	Treat	Treat	Treat	Treat	Treat
Number 4								
1	Treat	Treat	--	--	Treat	Treat	--	--
2	Abate	Abate	Treat	Treat	Abate	Abate	Treat	Treat
3	Treat	Treat	--	--	Treat	Treat	Treat	--
INST[a] No. 1	--	--	--	--	--	--	--	--
Level 2 streams								
Number 1								
1	Treat		Treat		Treat		Treat	
2	--		Treat		--		Treat	
Number 2								
1	Treat		Treat		Treat		Treat	
2	--		Treat and abate		--		Treat and abate	
3	Treat		Treat		Treat		Treat	
INST No. 2	--		--		--		--	
Number 3								
1	Treat	Treat	--	--	Treat	Treat	Treat	Treat
2	--	--	Treat	Treat	Treat	--	Treat	Treat
3	Treat	Treat	--	--	Treat	Treat	--	--
Level 3 streams								
Number 1								
1	--		--		Treat		Treat	
2	--		--		Treat		Treat	
3	--		--		Treat		Treat	
4	--		--		Treat		Treat	
5	--		--		--		--	
INST No. 3	--		--		--		--	
6	--		Treat		Treat		Treat	
7	Treat		--		Treat		Treat	

[a] INST = Instream treatment site

[b] -- = No action

Table 8. MINIMUM ANNUAL POLLUTION CONTROL COSTS FROM THE DWMC MODEL

	Annual Cost
Spring rates	
8 streams	$229,000
3 streams	84,000
Summer rates	
8 streams	259,000
3 streams	77,000

however, there is a different solution at two nodes for the maximum effectiveness model. Thus, the subsystem or sub-basin results and basin results are similar, but different solutions can occur. It should be noted that the case chosen would not be as sensitive to changes from a sub-basin to a basin because there are no upstream inputs to the sub-basin.

The differences between the results for spring and summer rates obtained from the DWMC model will present more problems in actual application. An examination of the DWMC computer runs indicates that the optimal solution based on the summer flow rates would not satisfy the five ppm quality standard using spring flow rates. The opposite comparison cannot be made without rerunning the computer program. This comparison would imply that "worst cases" are somewhat unique and a solution for one "worst case" is not necessarily adequate for another. This problem in applying the models needs further investigation.

The computer times from these runs indicates that the algorithms developed can be operated for systems of this size economically. The longest computer times were required by the maximum effectiveness model to solve the complete basin problem with summer flow rates. For that case, approximately two minutes of CPU time was used on Ohio State's IBM 370 Model 165 computer for a cost of thirty dollars. The computer times required by the DWMC model were about one-half of the DWME times. Thus, computer costs for these optimization models will be inexpensive for problems of this magnitude.

SECTION VI

EVALUATION OF OVERALL TECHNIQUE

The major contribution which has been made in the development of the overall resource allocation model described in the preceding sections lies in the specific capabilities of the two major sub-model types incorporated in the overall model; the source models, and the optimization models. The source models provide a rational means of describing and predicting the complex time dependent acid production phenomena which occur in the predominant types of mining environments. Until now, no such analytical tool has been available for simulating acid flow and load data. The optimization models, utilizing flow and load data, together with appropriate cost information, provide a compact and efficient cost optimization algorithm capable of determining the least-cost set of pollution control decisions for a branching array of acid sources, with the option of either drainage treatment or at-source abatement at each acid source and a given specific water quality standard. Alternatively, the optimization sub-model can determine the distribution of pollution control decisions over the total array of acid sources which will result in the most desirable water quality obtainable, given an upper limit on dollars available for pollution control. This sub-model also represents an analytical tool which has not been heretofore available.

Of the two sub-models, the source models will be the most difficult to apply realistically to field situations. The complexities inherent in the natural phenomena of pyrite oxidation and mine drainage formation are reflected in the multitude of parameters required for even a marginally accurate description of acid flows and loads by the source models, and the calibration of the source models to field conditions will never be an easy task. However, this should in no way be used as an excuse to rely on more simplistic approaches for the estimation of acid flows and loads, as such methods have consistently demonstrated their inability to provide useful answers. The output of the optimization models may be severely limited by inadequate acid flow and load data used as input to this model, and attempts to use the optimization model without high quality input data will have questionable value until the sensitivity of optimization model results to input data variations is clarified. The value of the resources being mined and of the resources being threatened by the effects of mining justify and demand a high degree of sophistication in the analysis and solution of mine drainage problems.

Although the optimization model structure is simpler, thus making it appear easier to apply, most field situations will be difficult to evaluate realistically because of this simpler structure. Situations are likely to be encountered that are unlike the set of assumptions selected in formulating this model. Many of the requirements for extending and enriching the optimization model will only become apparent when they are actually used; thus, further development should be coupled with actual use rather than attempting to cover every contingency.

The following discussions are intended to give a realistic evaluation of the strengths and weakness of the overall resource allocation model. Since the source models are coupled in series to the optimization model, with no feedback from the latter to the former, the two submodels will be evaluated separately.

SOURCE MODEL

The deep mine and the Combined Refuse Pile-Strip Mine (CRPSMM) Source Models have been specifically designed to provide a means of predicting time traces of the drainage flow and acid load from a given acid source, for any period of time desired, and under any presumed precipitation sequence.

At present, advances in the basic understanding of pyrite oxidation and acid transport mechanisms over the past decade, together with an already well developed science of hydrologic modeling, have made it possible to frame the source models in rational, mechanistic terms, thus avoiding the inherent and proven inadequacies of empirical or statistical predictions of mine drainage behaviors. The model parameters which have been used to describe acid formation, and water/acid movement are subject to rational interpretation. Many of these parameters are, to varying degrees, open to independent determination in the laboratory or in the field. Others can be determined only by calibration of the models to field data.

The source models are themselves composed of a hydrologic model coupled to an acid generation-acid transport model. In the interest of keeping the acid generation/transport sub-models as simple as possible, it was necessary to develop the two separate models, deep mine, and CRPSMM. While both describe the same phenomena, the physical differences between deep and surface mines made dual model development preferable to a highly complex generalized model. Of the two source models, only the Deep Mine Source Model has been tested in the field. Realistic testing of the CRPSMM model awaits the availability of more complete field data than were available during the course of this study. While the ability of the deep mine model to describe field observations with reasonable accuracy has been demonstrated, the CRPSMM model is offered only as a first generation model and it may require significant modification before it can be used as a practical tool.

Recalling that both source models utilize the same hydrologic model (the OSU version of the Stanford Watershed Model), remarks as to the strengths and weaknesses of this hydrologic model are in order. Many hydrologic models are currently in use for purposes ranging from flood plain delineation to low-flow water quantity prediction. When water quality is not a consideration, it is often possible to utilize a special purpose model of relatively simple form. For example, in flood studies a model specifically designed to reflect flood flows might be used. Such a special purpose model would normally have little or no capabilities for the accurate prediction of subsurface flows and low-flow conditions. In the analysis of mine drainage, however, critical acid concentrations may occur under either high- or low-flow conditions, and in all cases, acid concentrations and loads are determined by both subsurface and overland flow components. Even in refuse piles, much of the acid load is transported in subsurface flow at extremely high concentrations, and this transport requires accurate accounting of subsurface flows by the hydrologic model. Upon the appearance of highly acidic subsurface flow components at the ground surface, dilution by overland flow may provide a major moderating influence on acid concentrations in the stream, and accurate accounting of overland flow is a necessary characteristic of the hydrologic mode. Of the hydrologic models available, few are designed to provide a high degree of capability for both surface and subsurface component accounting. The Stanford model has demonstrated such capability, and was chosen for this reason. However, it must be recognized that this complex hydrologic model, together with its formidable array of descriptive parameters and input data requirements, also demands careful calibration to specific field conditions, with the availability of three years of stream flow data being considered necessary for optimum calibration of the model. While the possibility of hydrologic model simplification exists, only experience in application of the current form of the model to mine drainage situations will tell whether, and to what degree, such simplifications may be possible.

The acid generation and transport portion of the Deep Mine Source Model is described in detail in Appendix A of this report. In the simplified description of a mine according to this model, the coal seam is divided into discrete elements by using a three-dimensional rectangular coordinate system, and acid production, storage, and removal are calculated for all elements in a manner which ensures a complete mass balance for oxygen, oxidation products, and water moving within the system and its surroundings. To avoid unnecessary complexity in the model, many of the constants used in the mathematical description of oxidation rate, product transport, etc., are assumed to be constant for the entire system, or for an entire layer in the multilayer coal seam. For example, the coefficient DIP, used to calculate product removal from each element by gravity diffusion, is assumed to be a constant for the entire mine. Since DIF is defined as the fraction of acid formed in an element which is transferred each day to the next lower element, the application of the same value of DIF to all elements is an obvious but necessary simplification. Data simply do not exist to justify estimation of individual values of DIF to each element in the mine.

A second example is the assignment of the same value of reaction rate constant to all elements in a given layer or slice of the strata being modeled. While this is approximately correct, it cannot be expected to hold true except as an average effective value for the layer.

While numerous other examples of the assignment of "average effective" values to model parameters can be cited, the above is sufficient to make the point that the highly complex nature of the mine system is idealized to a high degree. As long as the model simulates historical records to an acceptable degree, a higher degree of sophistication is unwarranted. The validity or fallacy of the degree of simplification currently used in the model can be determined only by future experiences in applying the model to a range of geographical locations and a range of mine sizes. The model as it now stands has a sufficiently large number of constants, factors, etc., so that it is possible to fit the model to any set of field data which is reasonably complete. Whether this will be an empirical exercise in curve fitting, or calibration of a basically valid model reflecting the fundamental mechanisms controlling water movement and pyrite oxidation remains to be determined. Indications to date are that the Deep Mine Source Model is indeed valid, and that the various factors and coefficients used do reflect the basic phenomena they are intended to describe. It, thus, holds the promise of being a useful tool not only in the prediction of future acid loads from an existing mine, based on model calibration using historical data, but also in the prediction of acid loads after specific at-source abatement programs are carried out, such as partial flooding, mine sealing of various types, etc. Such predictions are possible only on the presumption that the intended abatement technique will affect various model parameters in a predictable way. Experience to date at the McDaniels Test Mine in Southeastern Ohio has indicated that the model has this capability, but further testing is desired.

A constraint on the current model is the fact that it is written for a single mine, having a given set of descriptive parameters. When the Deep Mine Source Model is to be applied to a complex of mines, the analyst has the choice of applying the model separately to each mine, or of aggregating the individual mines into a single effective mine, and handling the effluent from this effective mine as if it discharged at a single point. Alternately, the total discharge from an effective mine might be reallocated to the individual mine locations on the basis of individual mine areas, with or without consideration given to other factors.

The manner in which a complex of mines should be approached will depend largely on the degree of uniformity of characteristics expected among the mines in a given area, and the use to which the model is being put. For a basin study, it is likely that the aggregation scheme outlined above would provide sufficient simulation, with the exception that mines in different seams would probably require separate handling due to large probable differences in the hydrologic characteristics of the different seams. However, for at-source abatement design and simulation for specific mines, the individual mines would almost certainly have to be modeled separately.

The complexity of the system being modeled, together with the number of input parameters which must be determined or estimated for the calibration and operation of the Deep Mine Source Model, make it necessary that the analyst be closely familiar with mine drainage systems. The model in its present form cannot be approached as a black box system, and is not amenable to operation by personnel who are not well versed in the subject.

Many of the above comments relating to the acid production and transport component of the Deep Mine Source Model also apply to the corresponding component of the combined Refuse Pile-Strip Mine (CRPSMM) Source Model, described in detail in Appendix B. The major difference between the deep mine and refuse pile-spoil bank simulations is that the former utilizes only the underground flow components of the Stanford model in the simulation of acid transport, while the latter uses both surface and underground flow components in this capacity. The fact that the acid is produced at or near the surface in the strip mine-refuse pile case, rather than in a deep cavity, makes the problem of simulation simpler in some respects, but more difficult in others.

In the strip mine-refuse pile models, consideration must be given to underground flow phenomena. A large percentage of the total acid load leaving these sources of acid drainage come from, or are carried by, the underground water flow. This flow tends to dampen the effects of storm runoff from refuse piles unless the slopes are unexceptionally steep or the surface unusually compact.

In strip mined areas, over half the acid load (over an extended period) is carried by underground flow. This chronic flow from strip mines has a major effect on acid concentration in receiving streams during low flow periods.

A major difficulty in estimating input parameters for the CRPSMM model is that while coal seam characteristics are relatively constant throughout a given deep mine, the actively oxidizing pyrite in a strip mine or refuse pile represents disturbed material, at or near the surface, in which pyritic material is mixed to varying degrees with relatively inert materials from the overburden. A careful study of the area, including physical and chemical analysis of both surface and subsurface samples, is necessary to give a basis pyrite location and reactivity.

OPTIMIZATION MODEL

Given cost estimates and inputs from the source models, the Optimization Models were formulated to determine the optimal resource allocation to different point sources involving treatment and abatement alternatives. The initial formulation of the optimization problem showed that it is discrete, nonlinear, and stochastic. Moreover, pollution and stream flows vary as a function of time and are inherently stochastic principally due to precipitation patterns. To simplify the mathematics, the problem was solved by assuming that a single worst case could be identified. That is, a single condition could be identified where any resource allocation giving a particular stream quality for the worst case flows would never give lower quality at other points in time. With this assumption, a nonlinear discrete optimization algorithm has been derived and is ready for use for a complex of mines and streams as described in Section IV.

Problems are likely to be encountered in using this Optimization Model in several areas as outlined below:

1. The single worst case assumption has been shown to be false as shown by the example presented in Section V.
2. Pollution and stream flows are known to be stochastic.
3. The sensitivity of resource allocation to input data variations needs definition.
4. Situations are likely to be encountered which are unlike the mine and stream complex depicted in Section IV.

These problems are discussed in the following paragraphs.

The example in Section V involved two worst case flow situations where one situation was identified as a high flow situation during the spring and another was a low flow situation representing summer conditions. If the resource allocation deduced from spring flows would satisfy quality standards in the summer or vice versa, the existence of a worst case would be confirmed and the decision problem would be to decide which situation to use as the input to the model. However, the results show that a spring least cost solution may not meet quality standards in the summer and vice versa. Moreover, other "worst cases" may exist. Possibly year A may produce a worst case that gives a poor quality solution in year B. To ignore this problem invites the possibility of introducing pollution control action that is thought to guarantee good stream quality but does not exist in actuality.

The stochastic nature of stream flows and pollution flows introduces additional problems in the analysis, but recognition of the stochastic nature of flows may lead to increased understanding and efficiency in the pollution control process. For example, assume that the planner knows the cost of meeting quality standards with a 0.8 probability in a given year as well as the added cost to give a 0.95 probability; then the search for the absolute worst case may be regarded as unnecessary.

The facts that multiple "worst cases" exist and the flows are stochastic generate questions concerning the resolution and accuracy of input data required by the Optimization Models. The model has been shown to be sensitive to changes in input data values but little is known concerning the degree of sensitivity. How accurate do the input values from the pollution source models need to be to result in good decisions concerning pollution control actions? Can rough actual data values be and somehow give "ballpark" estimates of the effectiveness of pollution control actions?

Even with all of the above questions answered, actual situations will present new problems in applying the Optimization Models. For example, the Lake Hope area in Ohio has underground mines with multiple openings. Control costs vary with the number of openings sealed and these openings must be sealed in a particular order since the elevations of the openings vary. This decision concerning which openings to seal cannot be handled by the existing model. In addition, individual mines in the vicinity of the Clarion River in Pennsylvania drain into several different streams with different quality characteristics. What if mines drain into a lake or reservoir? These real situations may present problems in the application of the Optimization Model.

Although the above discussion indicates the existence of problems, the current Optimization Models represent a significant advance in the capability of analyzing mine drainage pollution problems. The current algorithm is efficient and presents a challenge to construct since the discrete nonlinear nature of the equations involved eliminates the use of available algorithms such as linear programming and integer linear programming.

SECTION VII

PUBLICATIONS

The following publications have resulted from research conducted under this project:

Ricca, V. and Chow, K., "Acid Mine Drainage Quantity and Quality Generation Model," presented at the Annual Meeting of the American Institute of Mining, Metallurgical and Petroleum Engineers, Chicago, Ill., Feb., 1973. Accepted for publication in the Transactions, Society of Mining Engineers of AIME, Salt Lake City, Utah, December, 1974.

Johnson, S., "Computer Simulation of Acid Mine Drainage from a Refuse Pile," Master of Science Thesis, Department of Civil Engineering, The Ohio State University, March, 1973.

Maupin, A., "Computer Simulation of Acid Mine Drainage from a Watershed Containing Refuse Pile and/or Surface Mines," Master of Science Thesis, Department of Chemical Engineering, The Ohio State University, August, 1973.

The following presentations were conducted using material derived from this research project:

Clark, G., Ricca, V., Smith, E., "Presentation of Project Results to EPA Personnel and Guests," Washington, D. C., Feb., 1974.

Ricca, V., "Hydrologic Modeling," guest lecture presented at the Third Short Course on the Hierarchical Approach in the Planning, Operation and Management of Water Resources Systems, Case Western Reserve University, Cleveland, Ohio, May, 1974.

Ricca, V., "Acid Mine Drainage Modeling," Environmental Engineering Seminar, The Water Resources Center, The Ohio State University, May, 1974.

Ricca, V., CE 820 Advance Hydrology, course lectures on modeling, The Department of Civil Engineering, The Ohio State University, Jan., 1974.

SECTION VIII

APPENDICES

Page

A. Deep Mine Pollutant Source Model 58

B. Refuse Pile-Strip Mine Pollutant Source Model 104

C. Optimization Model 183

D. Cost of Drainage Treatment 459

APPENDIX A

DEEP MINE POLLUTANT SOURCE MODEL

TECHNICAL DETAILS AND COMPUTER PROGRAMS

This appendix contains details on the pollutant source model as to its history, modifications, and linking mechanisms.

An in-depth discussion of their application, showing input parameter values and selection methodology, data assembly, and typical graphical and tabular outputs is included.

The section concludes with a set of procedures to operate the deep mine pollutant source model.

The material presented herein was taken as much as possible from publications (papers presented and Master of Science Theses) written as part of this project during the research period.

INTRODUCTION

Acid mine drainage is a serious water pollution problem, for its contaminants will eventually affect the quality of the receiving streams. According to the Appalachian Regional Commission (1969)*, 10,500 miles of streams have been polluted by mine drainage, and 70 percent of this acid pollution is accounted for by underground mines. Due to the severe damage to aquatic life, to recreational and industrial use, and to domestic water supply, more stringent mining and anti-pollution laws have been legislated and coal mine operators are required to treat the mine water discharges to meet the standards. For years now, abatement and treatment methods have been extensively studied. Progress reports presented in the Fourth Symposium on Coal Mine Drainage Research, 1972[1] suggest that a thorough investigation and understanding of the basin discharge is necessary in order to cope with the mine drainage problem. The extent of mine water discharge contamination can be considered as a function of the basin streamflow and acid generation load. A conservative treatment cost estimation these days is $0.40 per 1,000 gallons for water with 100 ppm iron, 500 ppm acidity. These figures need not include collection and pumping costs nor the cost of dispersing of chemical waste by-products. To achieve optimal abatement and treatment of mine drainage, predictions of the quantity and quality of mine water discharges are needed.

---

*References listed at the end of this unit pollutant source model.

Basin discharge is a continuous process and it can be considered as a macro-system. This system can be subdivided into micro-systems to describe the various mine discharge types in the basin. The mining activities could produce acid water discharges from deep mines, strip-mines, or associated gob piles.

In this appendix we will discuss only one micro-system, the drift (deep) mine type by looking at a single mine and its watershed. The total research project considers a multiple mining complex in an extensive stream basin.

The system (a mine and its watershed) will be studied and described by mathematical relationships which will be processed with the aid of a digital computer. Once the mine discharge is formulated as functions of mine water and acid generation within, continuous output of mine drainage to the receiving stream flow can be simulated. From these the watershed discharge and quality can be predicted.

Mine drainage simulation models can be useful in: predicting mine water quantity and quality, quantitizing the cost for treatment prior to discharge, evaluating the effects of abatements efforts, and above all, helping to cope with mine drainage pollution problems.

The model that is presented herein is a hybrid of computer programs for the hydrologic behavior of a watershed and for the generation of acid mine water. The former is structured on hydrologic cycle concepts and the latter on pyrite oxidation kinetics and oxidation product removal. The total model is programmed for the high speed digital computer (IBM 370/165).

The major inputs to the model are: climatological, watershed characteristics, and mine characteristics data. By the use of the hydrologic portion of the model the amount and timing of water that flows through the pyritic system is determined. From this information, the acid production portion of the model predicts acid load generation due to leaching, inundation, and gravity diffusion. The outputs from the total model include: average daily minewater discharge, the associated acid concentration or load, plus the average daily flow in the receiving streams or at basin outlets.

CONCEPTS OF THE SIMULATION MODEL

The simulation model consists of two major components: minewater flow and acid load. The former is related closely to the concept of hydrologic cycle, whereas the latter is based on the concept of pyrite oxidation kinetics and oxidation product removal. A clear understanding of these concepts is needed in formulating the mathematical models.

To study the workings of the total model we will look at the two major portions separately then show how they are joined. Before getting into the model components a brief review of the hydrologic cycle, reaction kinetics, and oxidation product removal may be helpful.

Hydrologic Cycle

Three zones of hydrologic events can be assigned to describe a hydrologic cycle. Figure A.1 shows a schematic diagram of the cycle as used by the model.

1. Upper Zone - This zone is above the soil surface. It is used to describe the activities at and after precipitation above the soil surface. The activities include interception, transpiration, evaporation, overland flow, surface detention and depression storage.

2. Lower Zone - This zone is the soil between the water table and land surface. It is used to describe the activities of infiltration from the upper zone, percolation, interflow, and the degree of soil moisture saturation.

3. Deep Lower Zone - This zone is the soil below the water table. It is used to determine groundwater flow to the stream, and to deep storage (or aquifer). Because moisture percolates through the cracks and crevices of the aquifer, a time delay occurs between moisture entering the aquifer and leaving the aquifer as minewater flow.

The hydrologic cycle is a continuous process. Precipitation falls on the upper zone and some of it will enter into the lower zone and deep lower zone. The balance will enter the upper zone (interception plus depression storage). To complete the cycle, evaporation and transpiration occur from all three zones.

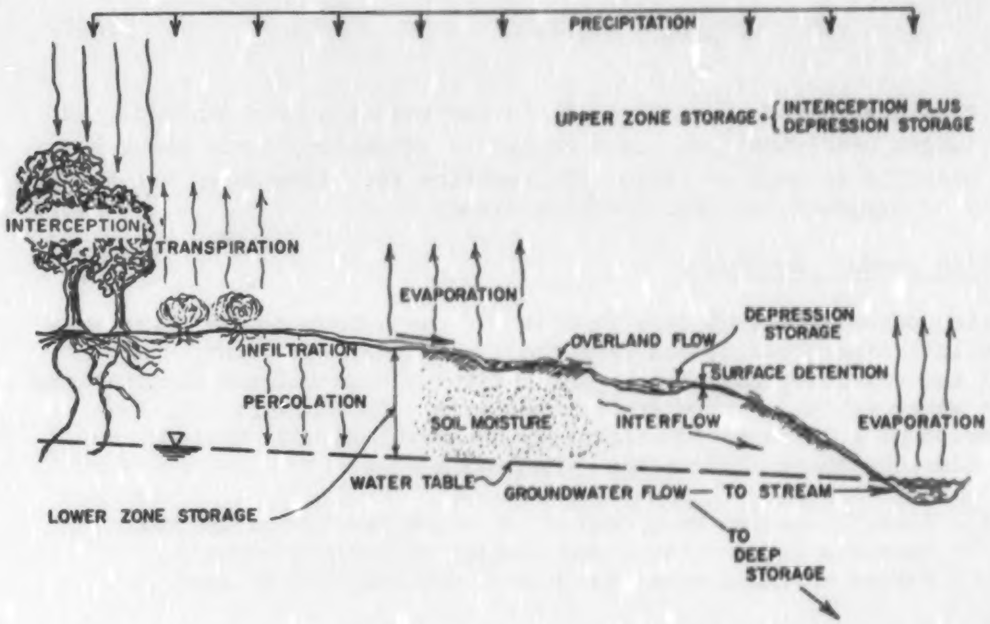

PRECIPITATION
UPPER ZONE STORAGE = {INTERCEPTION PLUS
DEPRESSION STORAGE
INTERCEPTION
TRANSPIRATION
EVAPORATION
DEPRESSION STORAGE
OVERLAND FLOW
SURFACE DETENTION
INFILTRATION
PERCOLATION
SOIL MOISTURE
INTERFLOW
EVAPORATION
WATER TABLE
GROUNDWATER FLOW — TO STREAM
TO DEEP STORAGE
LOWER ZONE STORAGE

Figure A.1. Schematic of hydrologic cycle

Pyrite Oxidation Kinetics

Acid mine drainage is caused by the natural formation of acid by the oxidation of iron pyrites (FeS_2 in the coal seams) in the presence of water and air. The reaction in its simplest form can be represented by equation A.1.

$$FeS_2 + 7/2\ O_2 + H_2O = H_2SO_4 + FeSO_4 \qquad (A.1)$$

The oxidation kinetics of pyrites have been thoroughly investigated.[2-5] Lau et al.[6] have reported that bacterial catalysis is not likely in underground environments. Smith and Shumate[7] have also shown that zero order reaction kinetics enhanced by microbial activity are not significant in underground pyritic systems. Morth and Smith have stated that the oxidation rate can be satisfactorily approximated as first order with respect to oxygen concentrations between the range of zero to 21 percent in mines. Equation A.2 shows the rate of oxidation:

$$r = k_r C_O \tag{A.2}$$

where r is the rate of oxidation, k_r is the reaction rate constant, and C_O is oxygen concentration. Because pyrite oxidations occur along the small channels in coal or shale, the reaction rate depends on the availability of oxygen at exposed reaction sites.

Oxidation Product Removal

Oxidation product removal is a function of the hydrogeologic characteristics including porosity and permeability of the overburden, sizes of cracks and crevices, position of the oxidizing material and fluctuations of the water table.[7] Three removal mechanisms (see Figure A.2) have been proposed and tested by laboratory observations and physical considerations.[8] These three mechanisms are:

1. direct leaching by ground water which percolates through channels and fractures and removes oxidation products formed at times when the channel was not full of water,
2. flooding of products from an inundated volume by a rising water table, and
3. gravity diffusion of saturated solutions of reaction products.

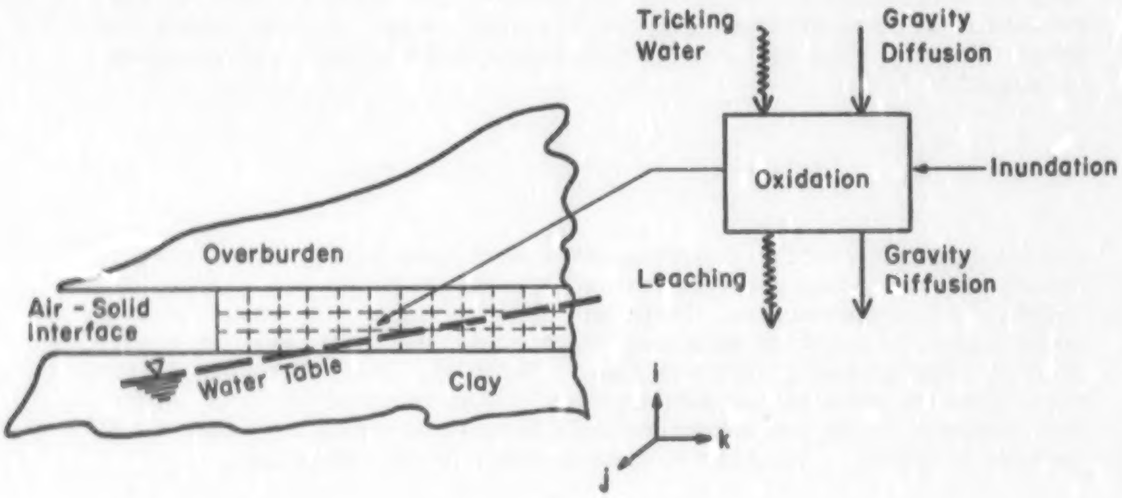

Tricking Water
Gravity Diffusion
Oxidation
Inundation
Leaching
Gravity Diffusion
Overburden
Air - Solid Interface
Water Table
Clay
i
k
j

Figure A.2. Underground pyritic system

The first two mechanisms which cannot be readily determined experimentally are closely related to underground water flow patterns, whereas the third mechanism which has been observed in laboratory studies is independent of water flow. Data of a high and a steady period of mine-water flow and acid load are needed to evaluate the parameters for the determination of the amount of acid removal by various mechanisms.

EVOLUTION OF THE MODEL

Once the concepts of the simulation have been established, mathematical models can be formulated to simulate the hydrologic cycle. The Ohio State University version of the Stanford Watershed Model with slight modifications is used to calculate the amount of moisture reaching the mine aquifer. A model described by Morth, Smith, and Shumate[9] can be used to simulate the pyrite oxidation kinetics and oxidation product removal. Evolution of these models are briefly described as:

Stanford Watershed Model (SWM)

The Stanford Watershed Model was developed by Crawford and Linsley in the early sixties.[10] James[11] modified the model and translated it into FORTRAN language in the late mid sixties. Various researchers[12,13] have worked with and proved the acceptability of the model in the late sixties. Since 1968 researchers at The Ohio State University have made modifications to progress the model along these lines: Balk[14] flow-diagrammed the model in detail and wrote an expose on the mechanics of its operation. Briggs[15] developed a computer plotted hydrograph program and made a sensitivity study of the key parameters. Owen[16] added multiple recession constants and a swamp and soil crack storage routines to the model. Mease[17] developed a snowmelt subroutine for the Midwest. Valentine[18] made modifications to make the model applicable to small watersheds. Warns[19] compiled a user's manual. The completed model in its present form has been summarized by Ricca and presented in a three-part report.[20] All programs can be executed by an IBM 370/165 time sharing computer system.

The original model consists of 15 input-output control options. The Ohio State University version has been extended to 20. Besides the climatological data (precipitation, pan evaporation, wind speed, solar radiation, and temperature) 31 input parameters (12 measurable parameters, 11 trial and adjustment parameters, and 8 assigned or selected parameters) are required by the model. Table A.1 lists the definitions, names, and sample values (see Case Study later in this appendix) of these variables. A detailed explanation to determine these variables is provided by Ricca.[20] The SWM is formulated by using these variables together with the concept of hydrologic cycle discussed in the previous section. A block diagram of the program for the model is shown in Figure A.3.

Table A.1. SWM PARAMETERS

Model Parameters	Parameter Definitions	Sample Value
	Measurable Parameters	
A	Impervious area that drains directly into the stream channel	0.00
AREA	Watershed drainage area in square miles	1.01
CHCAP	Index capacity of the existing channel in cubic feet per second	39.
COE	Empirical constant for convection	0.00177
ETL	Estimate of the stream and lake surface area as a fraction of AREA	0.001
IRC	Daily interflow recession constant	0.0313
K1	Long term ratio of average basin rainfall to average watershed ppt.	1.0
KK24	Daily baseflow recession constant	0.0226
KSC	Streamflow routing parameter for low flows	0.966
KSF	Stream routing parameter for flood flows	0.737
L	Mean overland flow path length in feet	1120.
SS	Average ground slope in feet/foot of the overland flow surfaces	0.0122
	Trail and Adjustment Parameters	
CB	Index controlling the rate of infiltration	2.2
CX	Index to estimate interception, depression storage capacity of the soil surface	0.7
CY	Index controlling time distribution, quantities of moisture entering interflow	3.0
EDF	Index for estimating soil surface moisture storage capacity	0.4
EF	Factor relating infiltration rates to evaporation rates for seasonal adjustment	4.0
EMIN	Minimum value of EN	0.1
GWS	Current value of groundwater slope index in inches	0.100
KV24	Daily baseflow recession adjustment factor	0.75
LZS	Current soil moisture storage in inches	4.5
LZSN	Soil profile moisture storage index, in inches	6.0
SGW	Groundwater storage increment, in inches	0.100
	Assigned or Selected Parameters	
EPXM	Maximum interception rate for dry watershed	0.2
K3	Soil evaporation parameter	0.3
K24L	Index for groundwater flow leaving basin	0.0
K24EL	Groundwater evaporation parameter	0.0
NN	Manning's n for overland flow on soil area	0.400
NNU	Manning's n for overland flow on impervious area	0.10
RFC	Index for routing	1.0
UZS	Current soil moisture storage	0.0

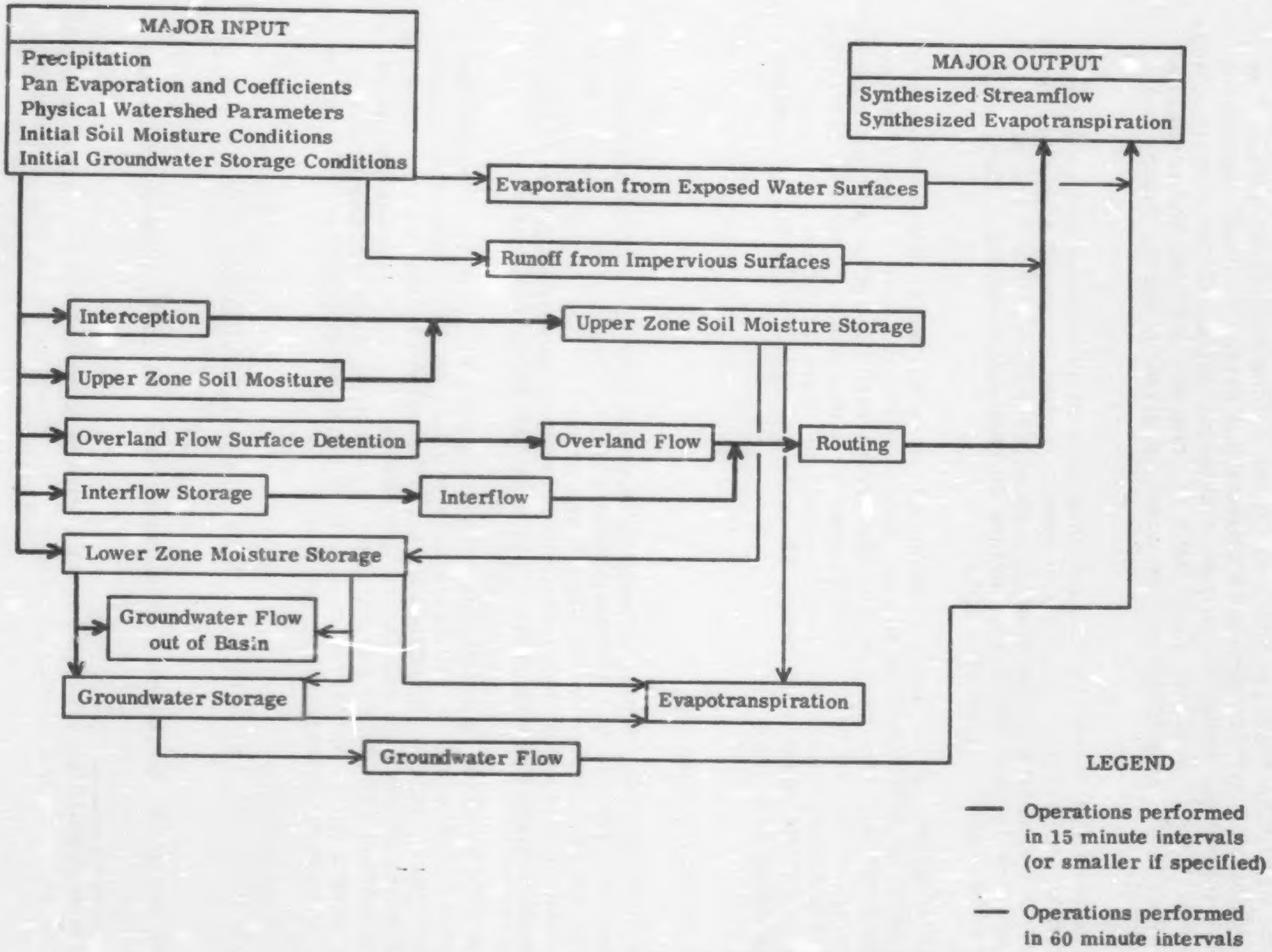

MAJOR INPUT
Precipitation
Pan Evaporation and Coefficients
Physical Watershed Parameters
Initial Soil Moisture Conditions
Initial Groundwater Storage Conditions
MAJOR OUTPUT
Synthesized Streamflow
Synthesized Evapotranspiration
Evaporation from Exposed Water Surfaces
Runoff from Impervious Surfaces
Interception
Upper Zone Soil Moisture Storage
Upper Zone Soil Mositure
Overland Flow Surface Detention
Overland Flow
Routing
Interflow Storage
Interflow
Lower Zone Moisture Storage
Groundwater Flow out of Basin
Groundwater Storage
Evapotranspiration
Groundwater Flow
LEGEND
Operations performed in 15 minute intervals (or smaller if specified)
Operations performed in 60 minute intervals

Figure A.3. Moisture accounting in Stanford Watershed Model

Acid Mine Drainage Model (AMD)

The Acid Mine Drainage Model was based on the conceptual model of an underground pyritic system by Shumate and Smith.[21] Chow[8] formulated the conceptual model to produce a mathematical model for a drift (deep) mine. Chow[22] utilized field data to progress and test the model. A more recent description of the model is given by Morth, Smith, and Shumate.[9]

The mine system can be divided into many micro-volumes and equations can be written to describe the events (oxidation and its product removal mechanisms) occurring in each micro-volume. The total mine behavior is then the sum of events in each micro-volume. Figure A.2 can be used to illustrate this.

In addition to the daily climatological data, twenty three parameters are to be determined to run the model. Nine are used to describe the mine system, five to describe the pyrite oxidation kinetics and nine to describe the oxidation product removal mechanisms. Table A.2 lists the definitions, names, and sample values of these variables. A block diagram of the model's oxidation kinetics and product removal mechanisms is shown in Figure A.4.

TOTAL MODEL (SWM-AMD)

In this section we will attempt to present only the basic modifications made to link the two aforementioned models. Readers who will be studying the combined model in detail are urged to study the individual models by reviewing the pertinent references mentioned for we will not backtrack unnecessarily over this material in this discussion.

The SWM involves calculations of hydrologic phenomena to describe the events that occur in the hydrologic cycle. From the lower zone, moisture percolates to the groundwater. The relationships for the fraction of moisture that percolates to groundwater (1-PRE)* and the degree of saturation (LZS/LZSN) in the lower zone can be shown in Figure A.5. Mathematically, infiltration moisture reaching groundwater (F1) can be expressed by equations A.3 and A.4.

$$F1 = (1.0\text{-}PRE)*(P4\text{-}SHRD)*(1.0\text{-}K24L)*PA \qquad (A.3)$$

for infiltration reaching groundwater from the lower zone storage, and

---

*These quantities are the actual computer program variable names and for the sake of consistency we will retain these names in this discussion.

Table A.2. AMD PARAMETERS

Model Parameters	Parameter Definitions	Sample Value
	Mine System	
WSHED	Watershed area of mine in square miles	0.00168
NFEET	Number of air-solid waterface increments	1
NLAYER	Number of layers in coal seam	10
NDEPTH	Number of depth increments in model	25
DK	Length of depth increments in feet	1
DI	Length of air-solid interface increments in feet	100.
TOP	Datum plane for top of coal seam in feet	13.8
ROCK, TYPE	Literal description of stratums	COAL
ALT	Elevation of stratum relative to datum plane in feet	10.0-13.8
	Pyrite Oxidation Kinetics	
REACT	Oxygen consumption rate of pyrite	2.55, 0.55
PYCON	Void fraction of the stratum	0.30; 0.005
TEMP	Mine temperature correction factor	0.15
FTGMOL	Volume occupied by gram mole gas in cubic feet	0.79
CCPRFT	Constant for calculating rate constant	28287.36
	Oxidation Product Removal	
TANK	Aquifer storage in inches	0.5
CONFH	Constant relating mine water flow and aquifer storage	0.0165
ALKALI	Alkalinity conversion factor	20.
FLOWMI	Minimum flow rate to cause acid removal by flooding	260.
HEADMI	Minimum flow rate to cause acid removal by leaching	0.00010
PER	Constant to determine the inundated distance	1.1
WSLOPE	Hypothetical slope of the water level	0.08
FRACT	Fraction of stored products removed daily by inundation	0.02
DIF	Base gravity diffusion constant	0.001

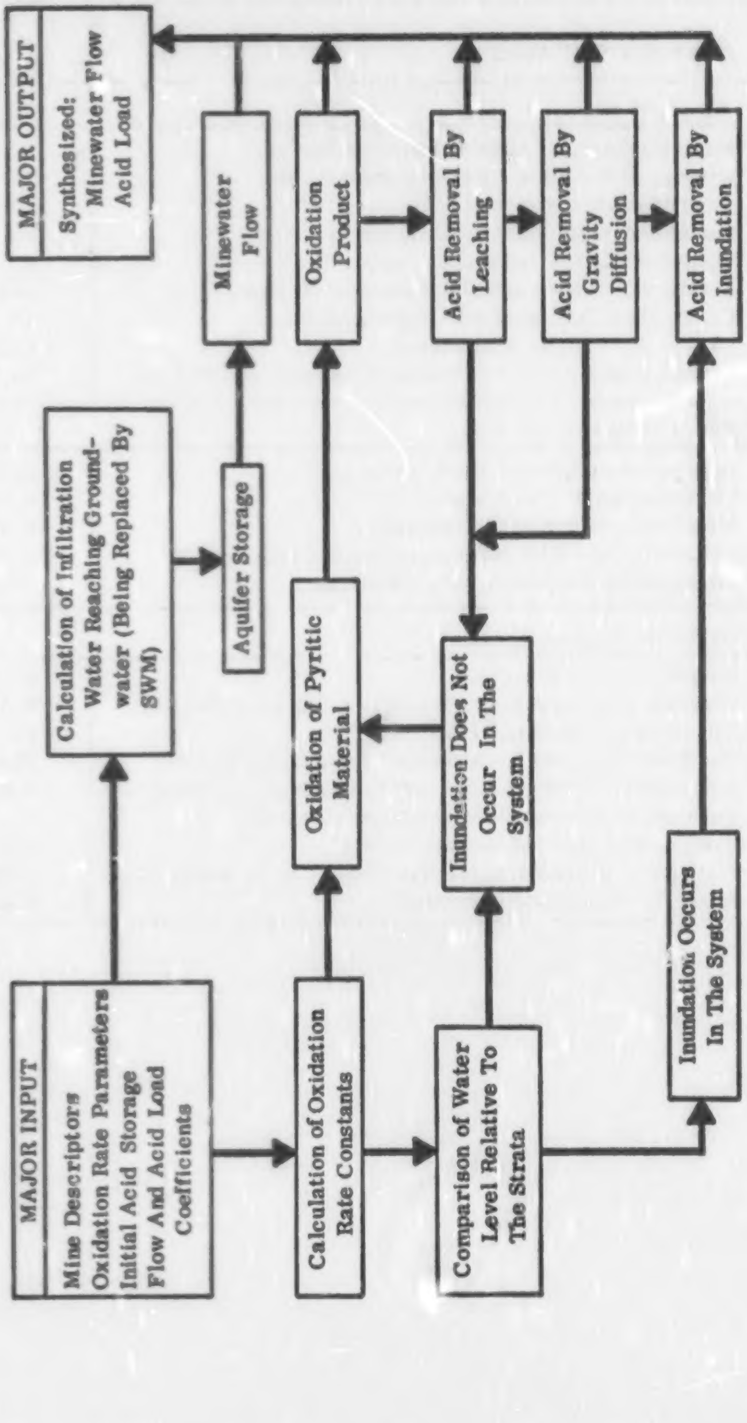

MAJOR INPUT
Mine Descriptors
Oxidation Rate Parameters
Initial Acid Storage
Flow And Acid Load Coefficients
Calculation of Infiltration Water Reaching Groundwater (Being Replaced By SWM)
Aquifer Storage
Minewater Flow
MAJOR OUTPUT
Synthesized:
Minewater Flow
Acid Load
Calculation of Oxidation Rate Constants
Oxidation of Pyritic Material
Oxidation Product
Comparison of Water Level Relative To The Strata
Inundation Does Not Occur In The System
Acid Removal By Leaching
Acid Removal By Gravity Diffusion
Inundation Occurs In The System
Acid Removal By Inundation

Figure A.4. Schematic of Acid Mine Drainage Model

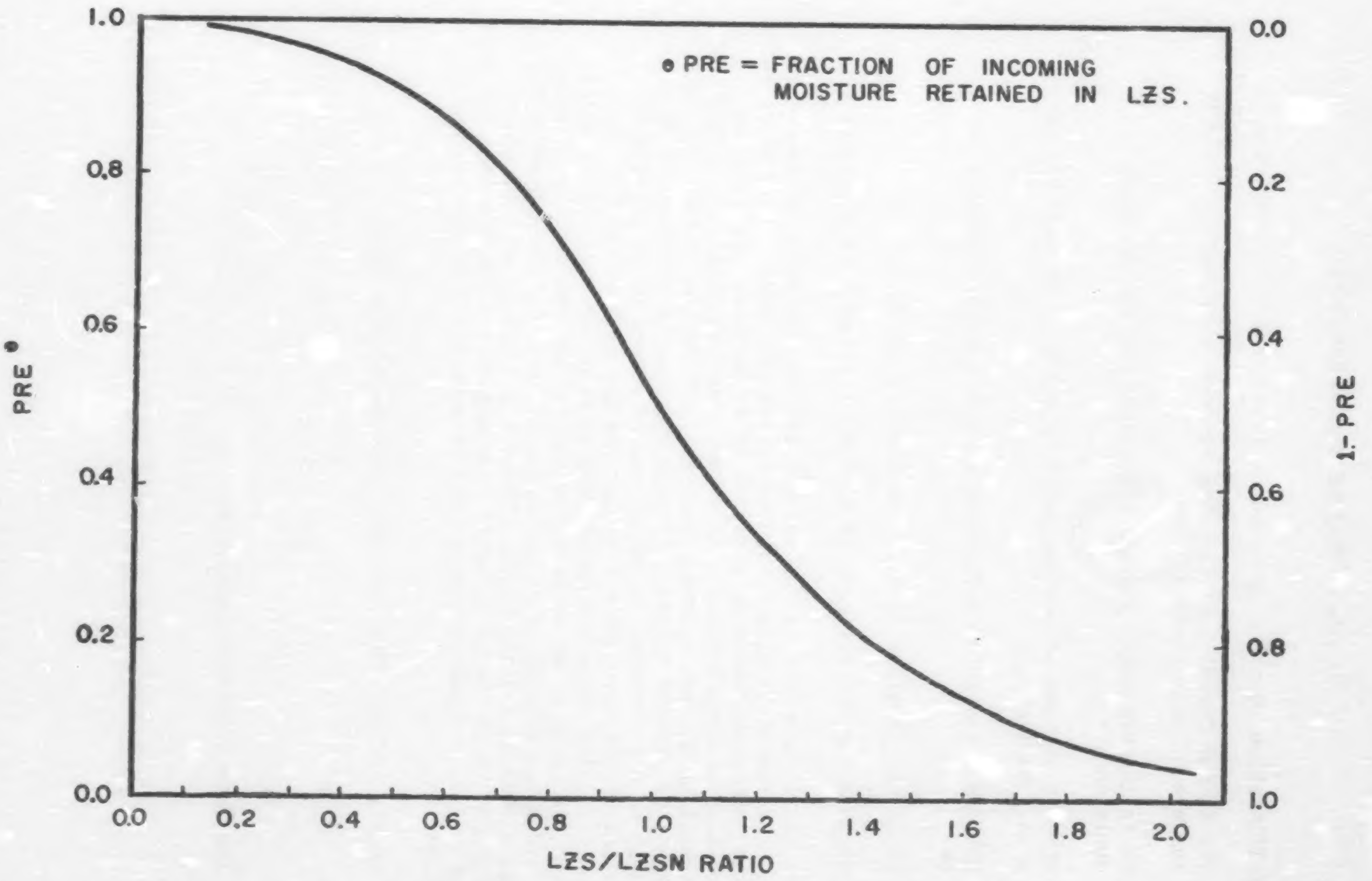


Figure A.5. Infiltration from UZS that is held in LZS

$$F1 = (1.0-PRE)*RECE*(1.0-K24L)*PA \tag{A.4}$$

for infiltration reaching groundwater from the upper zone, where

PRE is the fraction of incoming moisture retained in the soil surface or soil storage,

P4 is the residual rainfall after soil surface moisture depletion,

SHRD is the sum of current moisture entering surface runoff plus interflows,

K24L is a parameter indicating groundwater flow leaving the basin,

PA is the pervious fraction of the watershed, and

RECE is the current rate of soil surface moisture infiltration.

Values of F1 in equations A.3 and A.4 are calculated in 15 minute intervals. The sum of equations A.3 and A.4 is the total infiltration water reaching the groundwater in a period of 15 minutes. As shown in Figure A.1, the groundwater will either leave the basin as streamflow or it will go to deep storage (aquifer). It is the latter that enters the mine aquifer and trickles through channels in the pyritic system. These 15 minute interval F1 values can be summed up to provide the daily infiltration moisture reaching the groundwater (SADD). The reason for using daily values of SADD is that the time lag or delay between the groundwater entering an aquifer and the outflow of minewater from the aquifer is quite long and therefore the minewater flow varies slowly on a daily basis. These daily values of SADD can be punched out on IBM cards or written on magnetic tapes and then fed into the AMD model. These procedures can be summarized by the following formulations:

Hourly infiltration moisture reaching groundwater:

$$ADD = \sum_{1}^{4} F1$$

for four 15-minute intervals.

Daily infiltration moisture reaching groundwater:

$$SADD = \sum_{1}^{24} ADD$$

for 24 one-hour periods.

There are 365 (or 366) SADD values in a water year. Once the daily SADD values can be determined by the SWM, daily aquifer moisture storage (TANK) and daily minewater flow (FLOW) can be calculated by using the AMD model.[22] The relationships are:

$TANK = TANK + SADD(I)$ where $I = 1,2-----365$
$HEAD = f(TANK)$
$FLOW = f(HEAD)$
$TANK = TANK - FLOW$

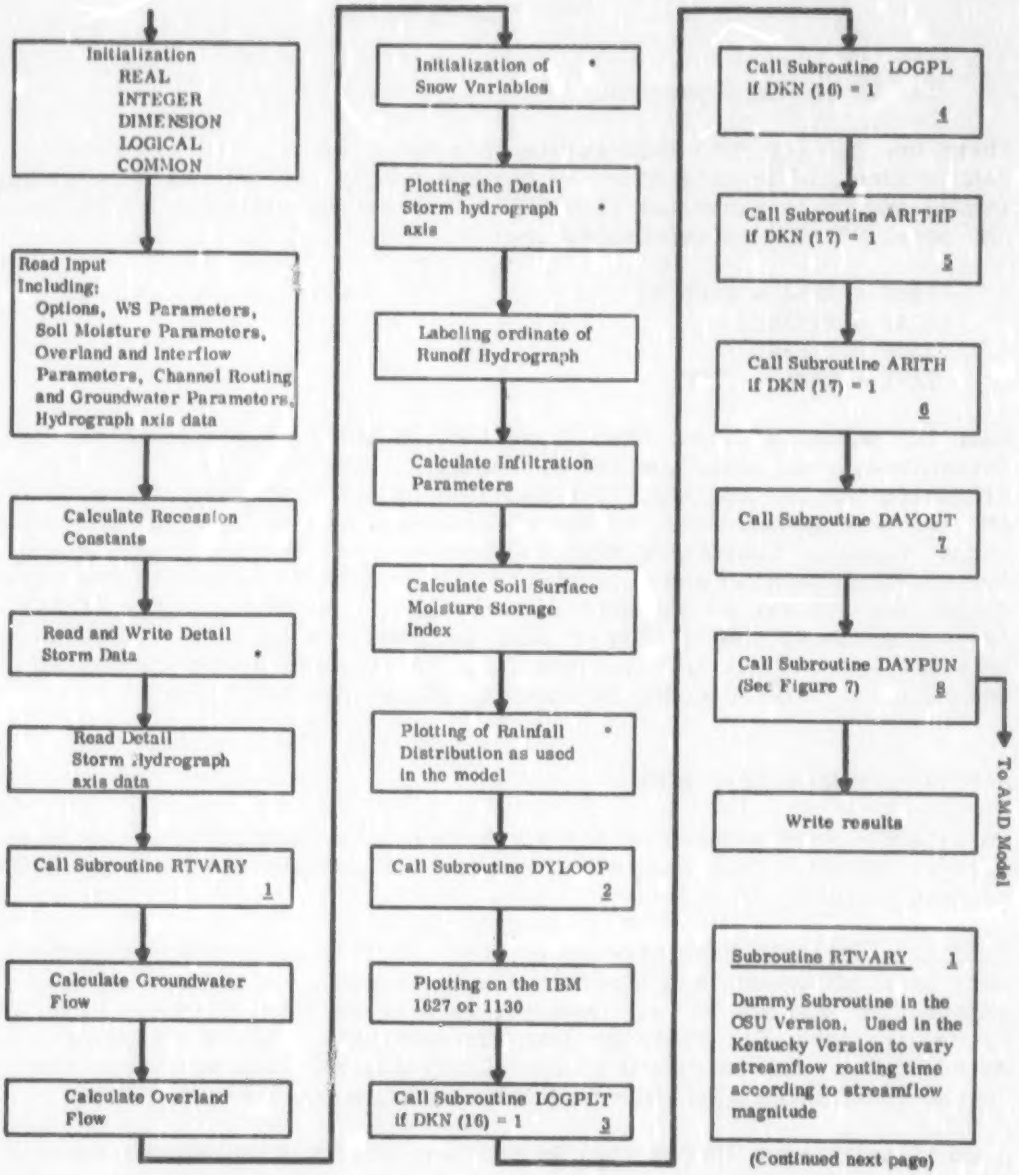
Once the amount of water that flows through the pyritic system can be determined, acid loads removed by leaching, inundation, and gravity diffusion can be obtained. The sequence of input parameters for the AMD has been modified due to the changes employed in linking the individual models. Subroutine MINE is added to read the input parameters to describe the mine and subroutine ACID is used to calculate the oxidation and removal of the pyritic materials. The MAIN program of AMD is to coordinate the linking of SADD and TANK, to calculate FLOW, and to write the outputs in tabulated form. A complete logic diagram to describe the SWM-AMD model is shown in Figure A.6 and Figure A.7.

APPLICATION TO A DEEP MINE

To determine the success of the simulation, the capability of the model must be tested. This suggests comparing the predictions from the model against existing data.

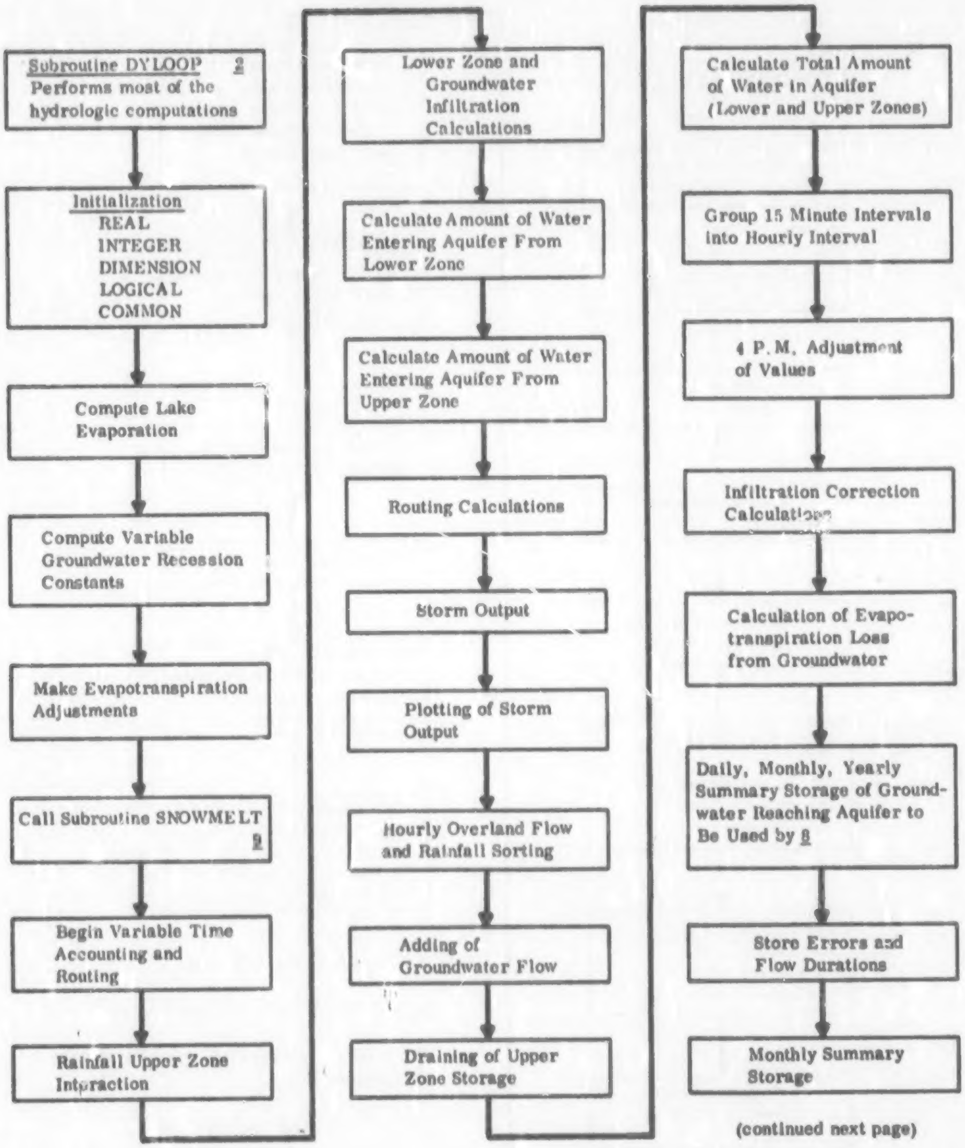
Finding a suitable test site poses great difficulty in that complete data on simultaneous hydrologic and acid minewater discharge are nonexistent or unknown to us. However, we were able to utilize a mine site where good mine drainage data was available, partial hydrologic data had been collected and enough climatological data was available in the immediate region to reasonably assemble the remaining data.

A small drift mine (McDaniels) in Southern Ohio was chosen for testing the model. The following will include a description of the mine watershed, a listing of the model input parameters developed for this site, the simulation output, and a discussion of the results obtained.


Initialization
REAL
INTEGER
DIMENSION
LOGICAL
COMMON
Read Input
Including:
Options, WS Parameters,
Soil Moisture Parameters,
Overland and Interflow
Parameters, Channel Routing
and Groundwater Parameters,
Hydrograph axis data
Calculate Recession
Constants
Read and Write Detail
Storm Data *
Read Detail
Storm Hydrograph
axis data
Call Subroutine RTVARY 1
Calculate Groundwater
Flow
Calculate Overland
Flow
Initialization of *
Snow Variables
Plotting the Detail
Storm hydrograph *
axis
Labeling ordinate of
Runoff Hydrograph
Calculate Infiltration
Parameters
Calculate Soil Surface
Moisture Storage
Index
Plotting of Rainfall *
Distribution as used
in the model
Call Subroutine DYLOOP 2
Plotting on the IBM
1627 or 1130
Call Subroutine LOGPLT
if DKN (16) = 1 3
Call Subroutine LOGPL
if DKN (16) = 1 4
Call Subroutine ARITHP
if DKN (17) = 1 5
Call Subroutine ARITH
if DKN (17) = 1 6
Call Subroutine DAYOUT 7
Call Subroutine DAYPUN
(See Figure 7) 8
To AMD Model
Write results
Subroutine RTVARY 1
Dummy Subroutine in the
OSU Version. Used in the
Kentucky Version to vary
streamflow routing time
according to streamflow
magnitude

(Continued next page)

Figure A.6. Logic diagram of The Ohio State University version of the Stanford Watershed Model


Subroutine DYLOOP 2
Performs most of the hydrologic computations
Initialization
REAL
INTEGER
DIMENSION
LOGICAL
COMMON
Compute Lake Evaporation
Compute Variable Groundwater Recession Constants
Make Evapotranspiration Adjustments
Call Subroutine SNOWMELT 9
Begin Variable Time Accounting and Routing
Rainfall Upper Zone Interaction
Lower Zone and Groundwater Infiltration Calculations
Calculate Amount of Water Entering Aquifer From Lower Zone
Calculate Amount of Water Entering Aquifer From Upper Zone
Routing Calculations
Storm Output
Plotting of Storm Output
Hourly Overland Flow and Rainfall Sorting
Adding of Groundwater Flow
Draining of Upper Zone Storage
Calculate Total Amount of Water in Aquifer (Lower and Upper Zones)
Group 15 Minute Intervals into Hourly Interval
4 P.M. Adjustment of Values
Infiltration Correction Calculations
Calculation of Evapotranspiration Loss from Groundwater
Daily, Monthly, Yearly Summary Storage of Groundwater Reaching Aquifer to Be Used by 8
Store Errors and Flow Durations
Monthly Summary Storage

(continued next page)

Figure A.6. Continued

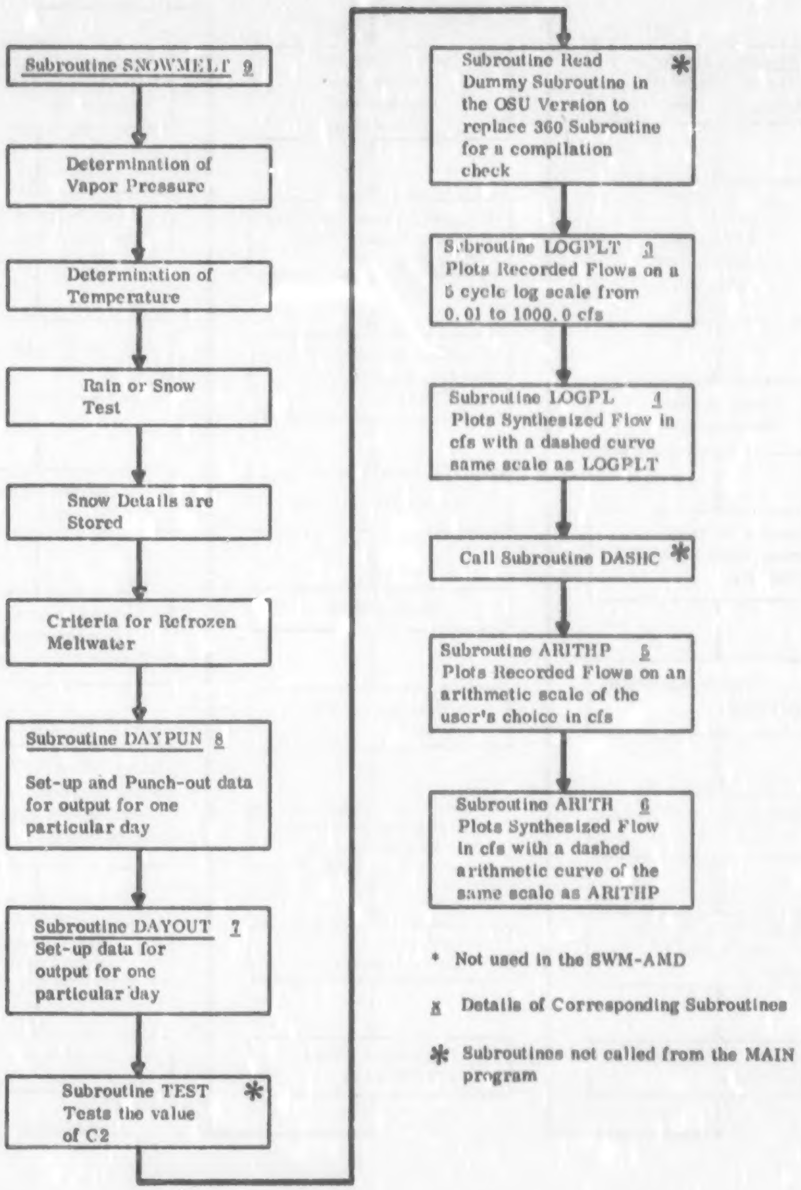
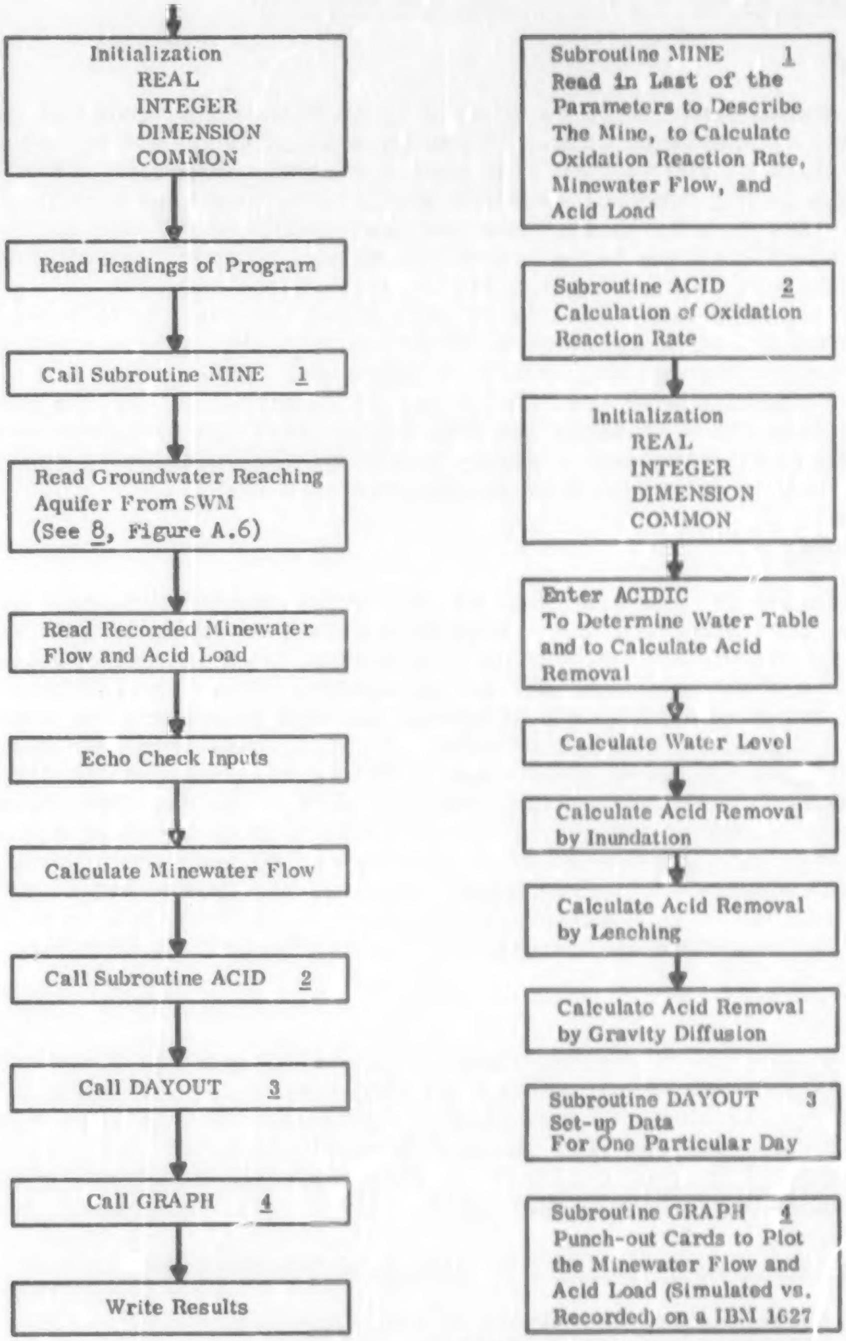

Subroutine SNOWMELT 9
Determination of Vapor Pressure
Determination of Temperature
Rain or Snow Test
Snow Details are Stored
Criteria for Refrozen Meltwater
Subroutine DAYPUN 8
Set-up and Punch-out data for output for one particular day
Subroutine DAYOUT 7
Set-up data for output for one particular day
Subroutine TEST ✱
Tests the value of C2
Subroutine Read ✱
Dummy Subroutine in the OSU Version to replace 360 Subroutine for a compilation check
Subroutine LOGPLT 3
Plots Recorded Flows on a 5 cycle log scale from 0.01 to 1000.0 cfs
Subroutine LOGPL 4
Plots Synthesized Flow in cfs with a dashed curve same scale as LOGPLT
Call Subroutine DASHC ✱
Subroutine ARITHP 5
Plots Recorded Flows on an arithmetic scale of the user's choice in cfs
Subroutine ARITH 6
Plots Synthesized Flow in cfs with a dashed arithmetic curve of the same scale as ARITHP
* Not used in the SWM-AMD
x Details of Corresponding Subroutines
✱ Subroutines not called from the MAIN program

Figure A.6. Concluded


Initialization
REAL
INTEGER
DIMENSION
COMMON
Read Headings of Program
Call Subroutine MINE 1
Read Groundwater Reaching Aquifer From SWM (See 8, Figure A.6)
Read Recorded Minewater Flow and Acid Load
Echo Check Inputs
Calculate Minewater Flow
Call Subroutine ACID 2
Call DAYOUT 3
Call GRAPH 4
Write Results
Subroutine MINE 1
Read in Last of the Parameters to Describe The Mine, to Calculate Oxidation Reaction Rate, Minewater Flow, and Acid Load
Subroutine ACID 2
Calculation of Oxidation Reaction Rate
Initialization
REAL
INTEGER
DIMENSION
COMMON
Enter ACIDIC
To Determine Water Table and to Calculate Acid Removal
Calculate Water Level
Calculate Acid Removal by Inundation
Calculate Acid Removal by Leaching
Calculate Acid Removal by Gravity Diffusion
Subroutine DAYOUT 3
Set-up Data
For One Particular Day
Subroutine GRAPH 4
Punch-out Cards to Plot the Minewater Flow and Acid Load (Simulated vs. Recorded) on a IBM 1627

x Details of Corresponding Subroutines

Figure A.7. Logic diagram of the Acid Mine Drainage Model

Description of the Study Area and Data Available

The Site -

The watershed under study is located in Southern Ohio, about 10 miles northwest of McArthur, Ohio. It has an area of about one square mile and a relief of 800 to 1040 feet mean sea level. The main stream in the basin is Big Four Hollow which drains into Lake Hope 4 miles downstream. The flow in this stream has been monitored for the last two years and will be the basic hydrologic data collected. Within the site, numerous deep mining activities were conducted. One of these deep mines, McDaniels Mine, is of particular interest in this study because Smith and Shumate[7] have developed this mine into a 'natural laboratory'. Through many years of continuous effort, flow and acidity data for the mine have been collected and reported. It is the purpose of this case study to apply the SWM-AMD and test its simulated values against the existing data. Figure A.8 shows the location of the study area in Ohio and the Big Four Hollow watershed area.

The Climate -

Precipitation in this area follows the general pattern of Ohio River Valley. Long duration, low intensity rainfall covering large areas occurs in the winter, whereas short duration, high intensity storms covering a small area dominate in the summer. Since the SWM requires precipitation data of hourly intervals and these data are not available for the exact location of McDaniels Mine, data from McArthur is used. Figure A.9 shows double mass plotting analysis used to test this data against three surrounding station's data. The McArthur record was judged to be representative of the precipitation at the mine site. Evaporation data for the study of this watershed were computed by using the meteorological data from nearby stations and processing it via the Penman method.[23] This method had been proven to work well in other Ohio areas therefore it was deemed adequate for this application.

The Geology -

The area consists of shales, clays, coal, sandstones and limestones. The mine is in the Middle Kittanning (#6) coal bed. The sandstone over the coal bed is about 40 feet thick. On the top of this is a thin layer of the Lower Freeport zone. The remaining upper part of the geologic section is of sandstone or silt composition with minor occurrences of shale, and thin coal seams.

Physical and Hydrologic Characteristics of the Watershed -

These characteristics are listed in Table A.3.

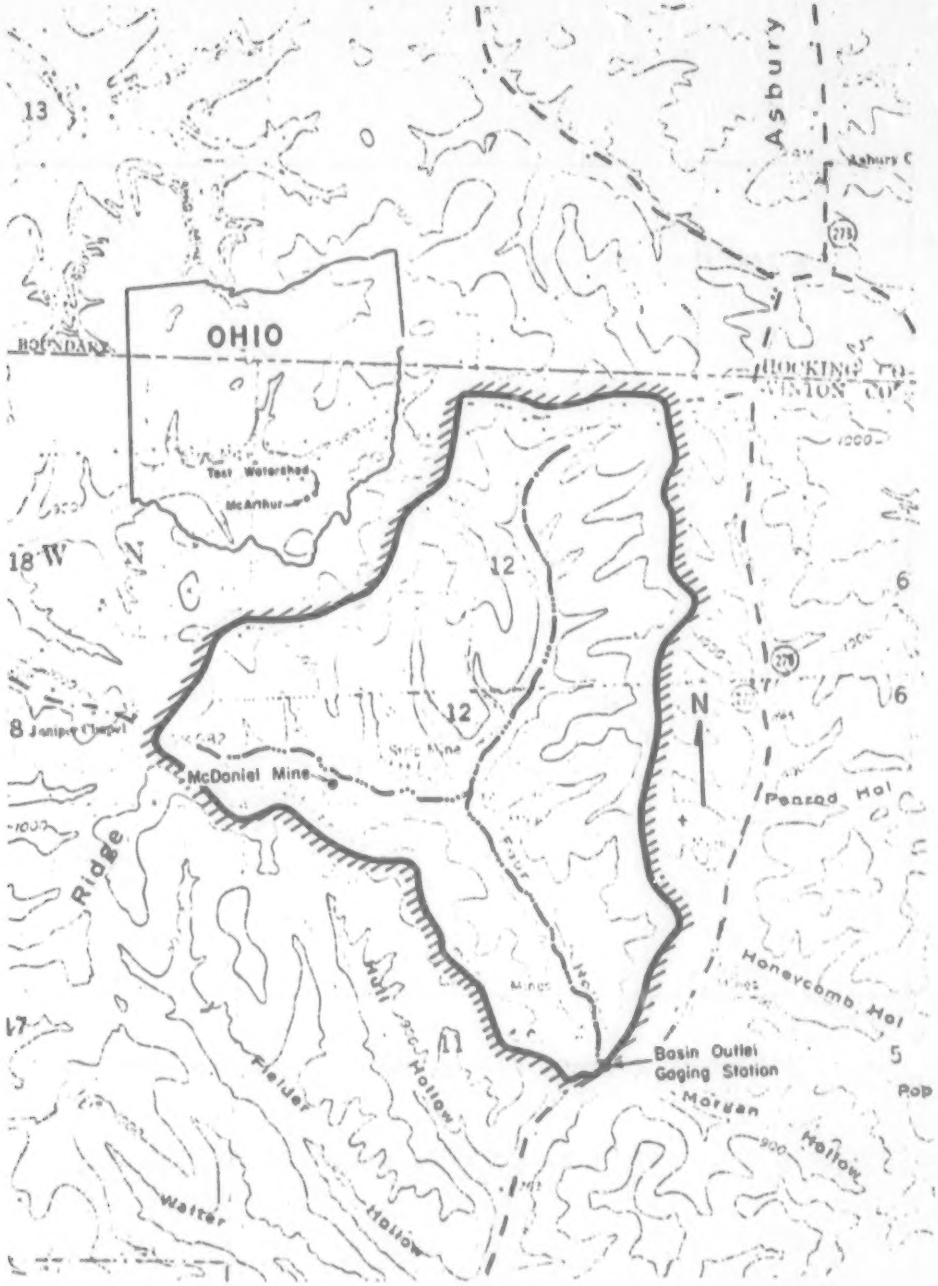

13
Asbury
Asbury C
273
BOUNDARY
OHIO
Test Watershed
McArthur
HOCKING CO.
VINTON CO.
1000
18 W
N
12
12
6
276
6
8 Juniper Chapel
Strip Mine
McDoniel Mine
N
Penrod Hol
Ridge
Honeycomb Hol
Mines
Basin Outlet
Gaging Station
5
Pop
Morgan
Hollow
Hollow
Fielder
Hollow
Walter
11
17

Figure A.8. The test watershed site

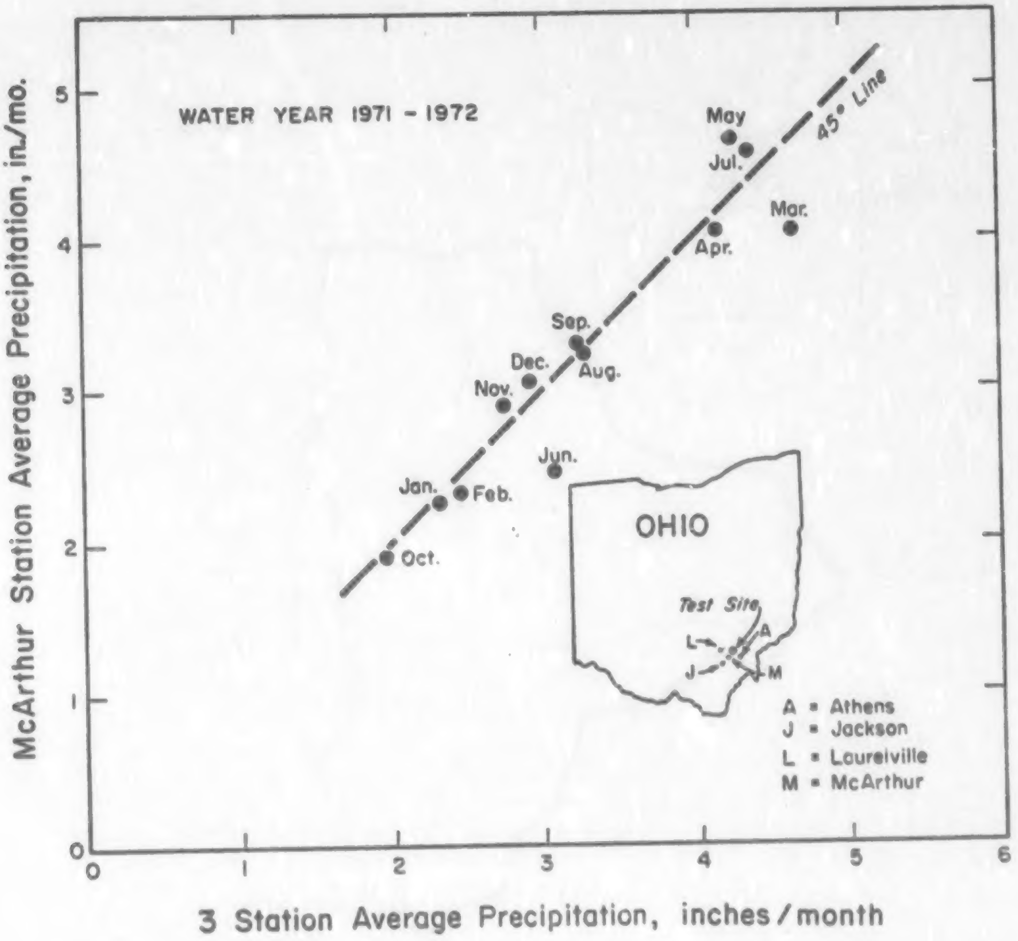


Figure A.9. Double mass plot of precipitation data

Table A.3. BASIN AND MINE CHARACTERISTICS

The Basin	Characteristics	
	Drainage area, sq. miles	1.01
	Length of principal water course, feet	1120.
	Average slope percent	0.0122
	Peak discharge of record, cfs	27 (2/22/70)
	Land use	heavy forest cover
The Mine	Area, sq. ft.	600.
	Average height, ft.	3.
	Peak minewater flow recorded, gallons per day	788 (5/27/68)
	Peak acid load recorded, lbs per day	1.9 (5/27/68)
	Principal materials in coal seam	Coal, shale

Mine Characteristics -

Also shown in Table A.3.

Program Input -

1. For Groundwater and Streamflow--Input parameters to calculate the groundwater reaching the mine aquifer and the streamflow are listed in Table A.1 according to their actual program name. See reference 20 for a detailed description of the variable names, their dimensions, and the methodology for obtaining their value.

2. For Minewater and Acid Load--Input parameters to calculate the minewater flow and acid load are listed in Table A.2 according to their actual program name. See reference 9 for a detailed description of the variable names, their dimensions, and the methodology for obtaining their values.

Program Output -

Simulation outputs are obtained both in tabular and graphical forms.

1. Daily infiltration water reaching the mine aquifer is listed in Table A.4. A hydrograph of the streamflow at the watershed outlet is shown in Figure A.10.

2. Daily minewater discharge and its acid load are listed in Tables A.5 and A.6 respectively. Their graphical output are shown in Figures A.11 and A.12. Monthly and annual summaries of minewater flow and acid load by component source generation are tabulated in Table A.7.

Table A.4. DAILY INFILTRATING WATER REACHING THE MINE AQUIFER

DAILY INFILTRATION WATER REACHING GROUNDWATER, BIG FOUR HOLLOW

WATER YEAR 1970-1971 UNIT IN INCHES

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT
1	0.0	0.002500	0.003100	0.002500	0.003400	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.007100	0.067600	0.002200	0.002000	0.003000	0.0	0.0	0.009200	0.0	0.0	0.0	0.0
3	0.0	0.008900	0.018500	0.007000	0.002700	0.0	0.0	0.0	0.0	0.0	0.027900	0.056500
4	0.0	0.005900	0.002700	0.054400	0.002400	0.0	0.0	0.0	0.0	0.0	0.030800	0.000300
5	0.0	0.004700	0.001700	0.004800	0.054300	0.0	0.0	0.0	0.0	0.036600	0.007400	0.0
6	0.0	0.003600	0.001500	0.002500	0.006300	0.031800	0.0	0.092100	0.0	0.000300	0.001000	0.0
7	0.0	0.002100	0.001200	0.002100	0.010900	0.009300	0.0	0.157200	0.0	0.0	0.0	0.0
8	0.0	0.000600	0.000900	0.001900	0.051500	0.003500	0.0	0.051500	0.0	0.0	0.0	0.0
9	0.0	0.000100	0.000700	0.001500	0.010100	0.031800	0.0	0.006300	0.0	0.0	0.0	0.0
10	0.006300	0.000100	0.000300	0.001100	0.005000	0.043600	0.0	0.000700	0.0	0.112100	0.0	0.0
11	0.002500	0.0	0.038000	0.001000	0.004200	0.004000	0.0	0.0	0.0	0.053100	0.0	0.0
12	0.000100	0.0	0.045700	0.000900	0.003400	0.003000	0.0	0.066900	0.0	0.003900	0.0	0.005900
13	0.0	0.0	0.004200	0.031300	0.019400	0.001700	0.0	0.085500	0.043000	0.000500	0.0	0.005700
14	0.170800	0.051700	0.003100	0.047300	0.003800	0.000600	0.0	0.002500	0.078600	0.0	0.0	0.0
15	0.007900	0.015000	0.001800	0.005300	0.003300	0.012600	0.0	0.000100	0.004600	0.024900	0.0	0.0
16	0.004600	0.004200	0.045200	0.003900	0.002800	0.000500	0.0	0.014000	0.001200	0.000100	0.0	0.081100
17	0.001600	0.003300	0.015700	0.009500	0.018700	0.000100	0.0	0.0	0.000100	0.0	0.0	0.000900
18	0.000300	0.002700	0.004500	0.004500	0.003900	0.0	0.0	0.0	0.0	0.0	0.0	0.000500
19	0.000100	0.002100	0.003600	0.003800	0.007900	0.016200	0.0	0.0	0.0	0.0	0.0	0.000200
20	0.036700	0.038300	0.002600	0.013500	0.010300	0.007100	0.0	0.0	0.0	0.0	0.0	0.036100
21	0.043200	0.004400	0.048100	0.003100	0.007000	0.000500	0.0	0.0	0.0	0.0	0.0	0.002400
22	0.001900	0.013400	0.053500	0.009500	0.065900	0.0	0.0	0.0	0.0	0.0	0.0	0.000300
23	0.001100	0.004300	0.057800	0.009800	0.008000	0.004900	0.0	0.0	0.0	0.0	0.0	0.0
24	0.000100	0.002500	0.007500	0.006300	0.003000	0.000100	0.0	0.0	0.0	0.003500	0.0	0.0
25	0.0	0.002000	0.010300	0.003300	0.002100	0.0	0.0	0.0	0.0	0.007000	0.102400	0.0
26	0.0	0.001700	0.006800	0.018600	0.001100	0.0	0.0	0.0	0.088000	0.0	0.120400	0.056500
27	0.0	0.001300	0.005300	0.003000	0.000500	0.0	0.0	0.0	0.001900	0.0	0.002500	0.000800
28	0.0	0.001000	0.004400	0.002700	0.0	0.0	0.0	0.0	0.007500	0.0	0.000500	0.000400
29	0.089600	0.016700	0.003800	0.032000		0.0	0.0	0.0	0.000100	0.0	0.0	0.0
30	0.077800	0.003700	0.003300	0.012600		0.0	0.0	0.0	0.0	0.0	0.0	0.0
31	0.010800		0.0	0.0		0.0		0.0		0.0	0.0	

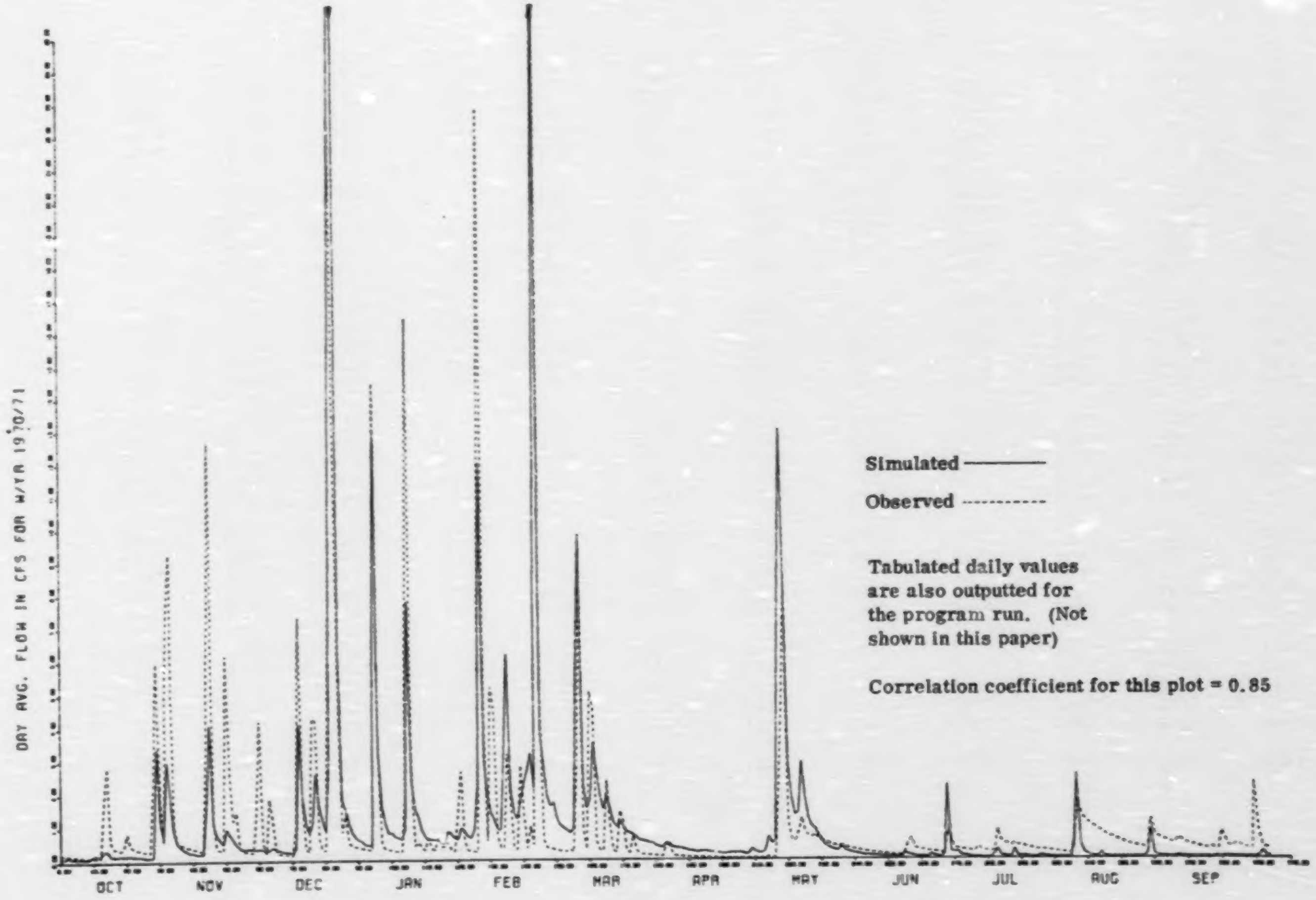


Figure A.10. Streamflow hydrograph at the Big Four Hollow watershed outlet

Table A.5. SYNTHESIZED DAILY MINEWATER DISCHARGE

SYN MINEWATER FLOW IN GALLONS FOR MINE # 1
WATER YEAR 1970- 1971

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT
1	220.	310.	252.	267.	239.	367.	276.	175.	260.	248.	233.	255.
2	219.	333.	248.	262.	236.	362.	271.	177.	256.	245.	229.	251.
3	216.	331.	251.	260.	232.	356.	267.	174.	253.	241.	238.	272.
4	213.	327.	247.	279.	229.	351.	263.	171.	249.	237.	248.	268.
5	210.	322.	243.	275.	248.	346.	259.	169.	245.	250.	247.	264.
6	206.	317.	239.	271.	246.	355.	256.	207.	241.	246.	244.	260.
7	203.	312.	235.	266.	245.	353.	252.	305.	238.	243.	240.	256.
8	200.	306.	231.	262.	263.	350.	248.	390.	234.	239.	237.	252.
9	197.	300.	226.	257.	262.	345.	244.	383.	231.	235.	233.	248.
10	197.	294.	222.	253.	259.	359.	241.	365.	227.	281.	230.	245.
11	195.	288.	234.	248.	256.	355.	237.	345.	224.	300.	226.	241.
12	192.	282.	249.	244.	252.	351.	233.	461.	220.	297.	223.	240.
13	190.	277.	246.	252.	256.	347.	230.	612.	236.	293.	220.	239.
14	262.	294.	243.	266.	252.	342.	226.	594.	267.	289.	216.	235.
15	261.	294.	239.	263.	391.	342.	223.	571.	265.	295.	213.	232.
16	259.	290.	254.	259.	386.	337.	220.	576.	262.	291.	210.	264.
17	256.	286.	255.	258.	389.	332.	216.	553.	258.	287.	207.	260.
18	252.	282.	252.	255.	385.	327.	213.	531.	254.	282.	204.	257.
19	249.	277.	249.	252.	382.	330.	210.	509.	250.	278.	201.	253.
20	261.	288.	245.	253.	381.	328.	207.	487.	246.	274.	198.	265.
21	276.	284.	261.	249.	378.	323.	204.	465.	243.	270.	195.	262.
22	273.	284.	280.	248.	402.	318.	201.	444.	239.	266.	192.	258.
23	269.	281.	299.	247.	399.	316.	198.	423.	235.	262.	189.	254.
24	265.	276.	297.	245.	395.	311.	195.	403.	232.	259.	186.	251.
25	261.	272.	295.	241.	390.	306.	192.	382.	228.	258.	228.	247.
26	257.	267.	292.	245.	384.	302.	189.	362.	264.	255.	278.	268.
27	253.	262.	289.	241.	379.	297.	186.	343.	260.	251.	274.	264.
28	250.	257.	285.	238.	373.	293.	183.	323.	260.	247.	271.	260.
29	285.	260.	281.	247.		288.	180.	304.	256.	243.	267.	257.
30	315.	256.	277.	247.		284.	178.	286.	252.	240.	263.	253.
31	315.		271.	242.		280.		267.		236.	259.	

Table A.6. SYNTHESIZED DAILY ACID LOAD

SYN TOTAL ACID LOAD IN POUNDS/DAY FOR MINE # 1
WATER YEAR 1970- 1971

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEPT
1	0.013	0.117	0.114	0.148	0.138	0.384	0.220	0.138	0.313	0.211	0.225	0.252
2	0.024	0.125	0.113	0.146	0.138	0.380	0.214	0.137	0.290	0.210	0.222	0.251
3	0.032	0.131	0.112	0.143	0.137	0.368	0.209	0.136	0.272	0.209	0.221	0.258
4	0.038	0.134	0.112	0.145	0.136	0.358	0.204	0.135	0.256	0.208	0.222	0.264
5	0.044	0.136	0.111	0.146	0.138	0.349	0.198	0.134	0.244	0.209	0.223	0.266
6	0.048	0.138	0.110	0.146	0.139	0.343	0.194	0.139	0.233	0.209	0.224	0.265
7	0.051	0.138	0.109	0.146	0.140	0.338	0.189	0.168	0.224	0.209	0.223	0.263
8	0.053	0.138	0.108	0.144	0.144	0.333	0.186	0.238	0.216	0.209	0.223	0.260
9	0.055	0.136	0.107	0.142	0.148	0.329	0.182	0.294	0.210	0.208	0.222	0.258
10	0.056	0.135	0.106	0.140	0.149	0.328	0.179	0.324	0.204	0.222	0.220	0.255
11	0.058	0.132	0.106	0.138	0.150	0.326	0.176	0.334	0.199	0.239	0.219	0.253
12	0.059	0.130	0.108	0.136	0.150	0.324	0.174	0.368	0.195	0.252	0.217	0.250
13	0.059	0.127	0.109	0.136	0.151	0.321	0.171	0.446	0.194	0.262	0.215	0.249
14	0.065	0.128	0.110	0.138	0.151	0.314	0.169	0.507	0.200	0.267	0.213	0.246
15	0.070	0.129	0.110	0.139	0.208	0.308	0.167	0.545	0.205	0.274	0.211	0.244
16	0.073	0.129	0.112	0.139	0.252	0.302	0.164	0.575	0.208	0.280	0.209	0.252
17	0.076	0.128	0.113	0.139	0.288	0.297	0.162	0.589	0.208	0.281	0.206	0.255
18	0.078	0.127	0.115	0.139	0.316	0.293	0.160	0.597	0.208	0.281	0.204	0.256
19	0.079	0.125	0.115	0.138	0.338	0.289	0.158	0.594	0.207	0.278	0.202	0.257
20	0.082	0.126	0.116	0.138	0.355	0.286	0.157	0.584	0.206	0.275	0.200	0.263
21	0.086	0.126	0.118	0.138	0.359	0.281	0.155	0.567	0.205	0.271	0.197	0.266
22	0.089	0.126	0.124	0.138	0.357	0.276	0.153	0.546	0.203	0.266	0.195	0.265
23	0.091	0.126	0.133	0.137	0.372	0.272	0.151	0.526	0.202	0.261	0.193	0.265
24	0.092	0.125	0.140	0.137	0.383	0.266	0.150	0.504	0.200	0.255	0.191	0.263
25	0.093	0.124	0.145	0.137	0.391	0.261	0.148	0.504	0.198	0.250	0.197	0.262
26	0.092	0.122	0.150	0.137	0.397	0.256	0.146	0.490	0.205	0.246	0.217	0.269
27	0.092	0.120	0.151	0.137	0.393	0.250	0.145	0.466	0.208	0.242	0.232	0.274
28	0.092	0.118	0.153	0.136	0.388	0.244	0.143	0.434	0.210	0.238	0.242	0.274
29	0.097	0.117	0.153	0.137		0.238	0.141	0.403	0.211	0.234	0.249	0.272
30	0.105	0.115	0.152	0.138		0.232	0.140	0.371	0.211	0.231	0.252	0.271
31	0.113		0.150	0.138		0.225		0.340		0.228	0.252	

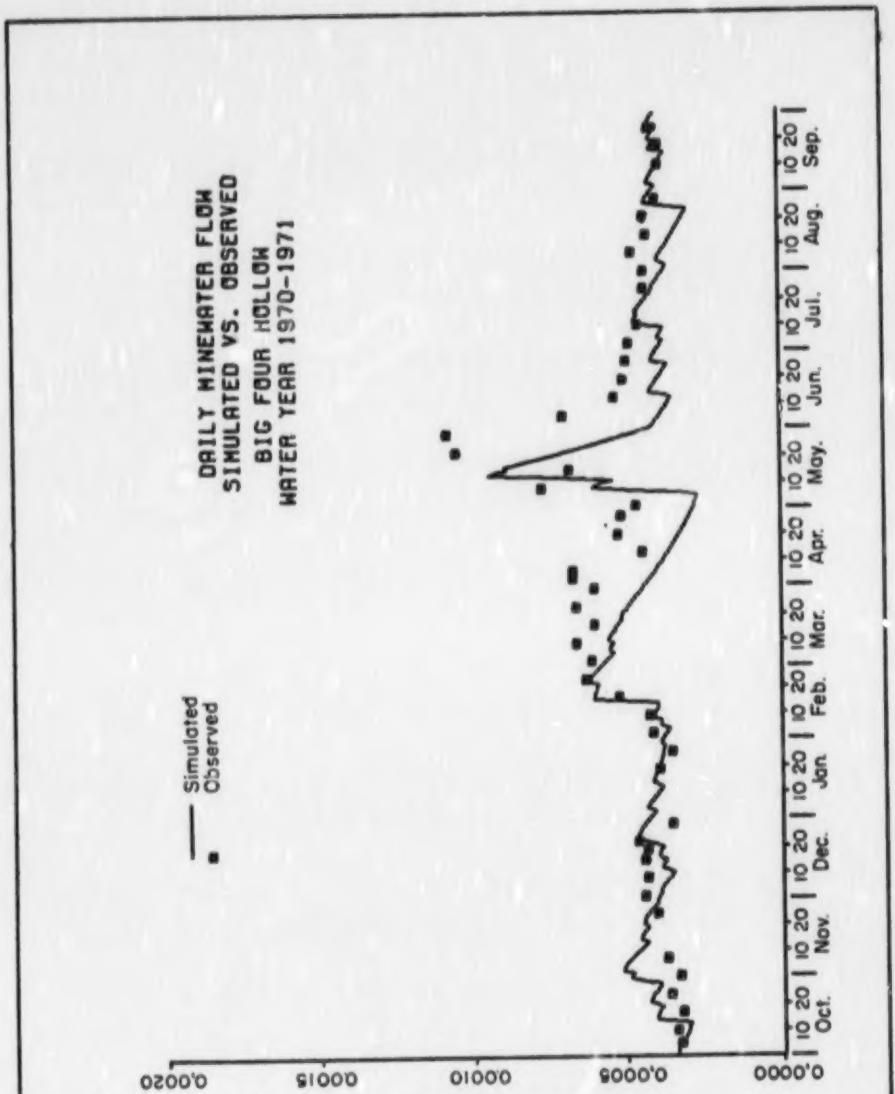


Figure A.11. Daily minewater discharge from McDaniel's Mine

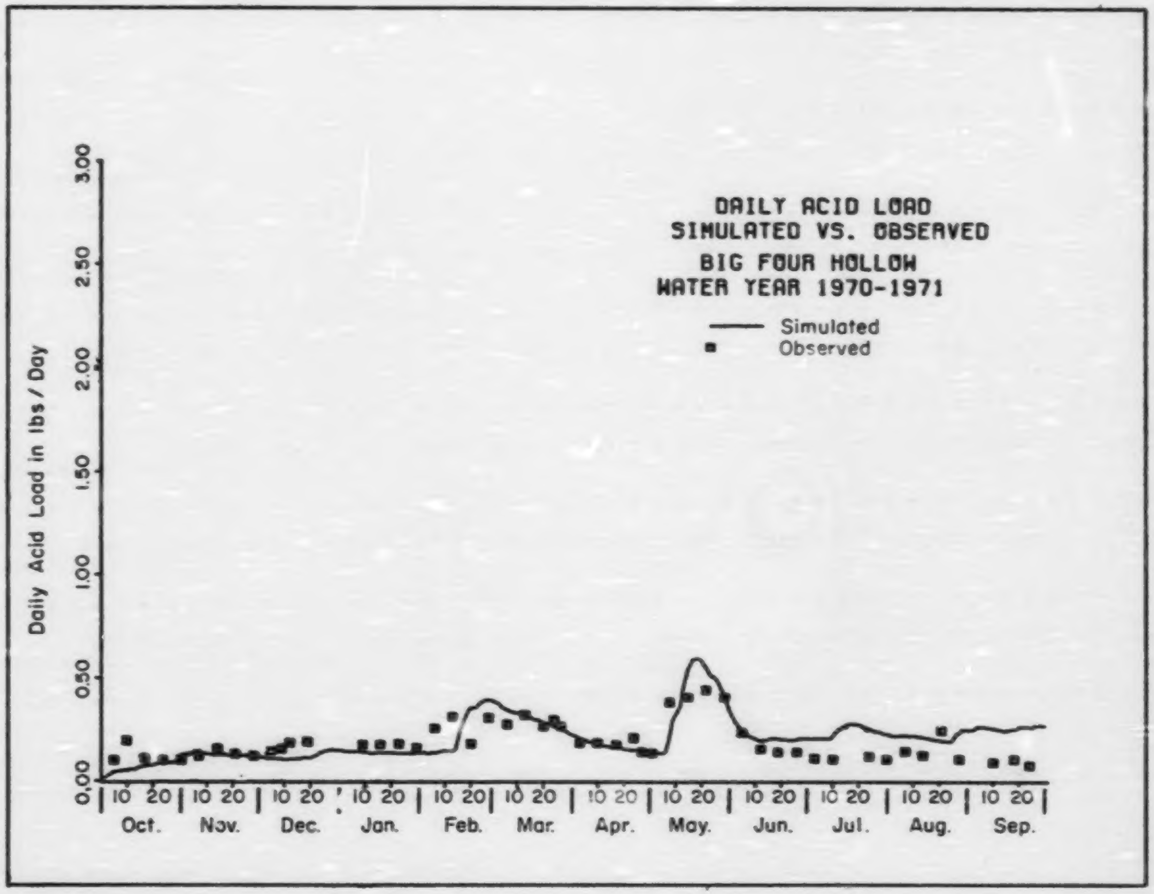


Figure A.12. Daily acid load discharge from McDaniel's Mine

Table A.7. SYNTHESIZED MONTHLY AND ANNUAL MINEWATER DISCHARGE AND ACID LOAD

```
ANNUAL SUMMARY FOR WATER YEAR 1970-1971     MINE # 1


                          OCT    NOV    DEC    JAN    FEB    MAR    APR    MAY    JUN    JUL    AUG    SEP


* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. INFILTRATION WATER REACHING GROUNDWATER                                                                *
                         0.462  0.264  0.398  0.296  0.315  0.141  0.0    0.486  0.225  0.242  0.293  0.248      3.371  INCHES

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. MINE WATER FLOW                                                                                        *
                        0.0115 0.0134 0.0123 0.0122 0.0137 0.0158 0.0103 0.0181 0.0114 0.0126 0.0109 0.0118     0.1542  CFSD

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. MINE WATER FLOW                                                                                        *
                          7480.  8708.  7987.  7895.  8888. 10256.  6697. 11757.  7383.  8136.  7093.  7631.     99910.  GALLONS

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. TOTAL ACID LOAD                                                                                        *
                            2.2    3.8    3.8    4.3    6.8    9.4    5.1   12.1    6.5    7.5    6.7    7.8       76.1  LBS

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. ACID LOAD BY LEACHING                                                                                  *
                            2.8    3.8    3.7    4.0    5.2    6.4    4.0    8.6    5.2    6.2    5.4    6.4       61.8  LBS

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. ACID LOAD BY GRAVITY DIFFUSION                                                                         *
                            1.0    1.0    1.3    1.5    1.3    1.5    1.8    1.5    2.0    2.2    2.4    2.5       20.2  LBS

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *

*    SYN. ACID LOAD BY INUNDATION                                                                                *
                            0.1    0.4    0.2    0.1    2.8    2.5    0.0    4.8    0.1    0.5    0.1    0.2       11.9  LBS

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Discussion of Results

The general trends for the streamflow, minewater flow and acid load for the single water year tested are reasonably well simulated. Because streamflow and minewater flow are based on hydrologic cycle concepts and acid load generation is based on the concept of oxidation and product removal mechanisms, the first two outputs are discussed together and the latter one separately.

Groundwater is the source of moisture supply to the minewater flow and a contributing source to the streamflow. The amount of groundwater available is determined by the hydrologic cycle modeling of SWM. Climatological data (extensive if snowmelt is involved) are needed to use the model. Generally speaking, these data are important to decide the accuracy of the simulation and they are usually scarce in isolated mining sites. In order to utilize the model, it is suggested that the availability of data should be checked. If it is not readily at hand, efforts should be made to collect these data a priori, if reliable predictions of mine drainage are to be obtained. In the present case study, the climatological data were for the most part approximated. For example, precipitation was estimated by utilizing three nearby stations' data. Furthermore, pan evaporation data are important to determine the evapotranspiration of the area, and again, these data were synthesized in this case study.

Besides climatological data, model parameters such as EF, EMIN, CB, EDF, are very sensitive to the values of groundwater. Numerous trials and adjustments are performed to choose their best values as judged by comparing simulation results to flow records. It is essential to note that all these parameters are interrelated so that it is necessary to adjust each one separately and check the effects on the groundwater behavior. A general guideline is to produce sufficient SADD which is required to simulate the minewater flow (as a function of aquifer storage) while at the same time match the recorded streamflow as closely as possible.

In the present case study only one year of data was used. This presents a problem in achieving equilibrium in the soil moisture balance in the watershed. The reports on the model state that at least three years of modeling are recommended for the adjustment period. Hence our single year of data falls short of this requirement.

The results of streamflow indicate a trend of undersynthesis in spring and oversynthesis in summer and fall. The under/or oversynthesis can be modified by adjusting the parameters such as EF, EMIN, CB, EDF to improve the simulation of streamflow. However, on the other hand, values of SADD will be too small or too large due to the adjustments. No definite explanation will be attempted at this time for these behaviors since the data used is too limited. A possible explanation to the undersynthesis is that snowfall during winter is unable to infiltrate into the frozen soil so that the modeled groundwater soil moisture in spring will not be representative of the actual watershed conditions. Again, the model is capable of handling the snowmelt problem but the data was not available. Possible explanations to the oversynthesis are: local rain storms occur in summer and fall seasons which are not represented correctly by the approximated precipitation data; evapotranspiration values are not large enough for the two seasons; and actual soil moisture may be considerably lower than the modeled volume during these seasons due to dry summer conditions.

The simulation of minewater flow is mainly based on values of SADD and TANK. These two values were optimized to give the best simulated results. However, in the spring the hydrologic model appears to be undersynthesizing the amount of water reaching the mine aquifer, that is, groundwater allocation, and this in turn causes the minewater discharge to be correspondingly undersynthesized. This adding of small or zero increments of SADD in the spring causes a reduction in the aquifer storage value, the effects of which propagate undersynthesis of mine flow into the early summer months.

The trend for acid load closely follows that for the minewater discharge because two of the removal mechanisms (leaching and inundation) are functions of the minewater flow. On the other hand, the gravity diffusion mechanism is relatively constant throughout the year. The major contribution to the sustained acid load removal is the leaching. Inundation accounts for the peak loads.

Generally speaking, the results obtained in this limited data study can not be used to justifiably evaluate the model's ability. Both of the individual models have been shown to work well with the proper quantity of data. At least two more years of data are needed to permit the model to reach equilibrium status. Also better climatological data including snow conditions should also improve the simulation.

In conclusion, taking all of the above discussion into consideration it is strongly felt that the model will be capable of predicting the minewater discharge quantity and quality.

PROCEDURES TO RUN THE DEEP MINE POLLUTANT SOURCE MODEL COMPUTER PROGRAM

Since this model was created by linking two previously established and published models, the program listing of both of these models will not be duplicated herein. They are available in the references (20) for the hydrology and (22) for the acid generation.

The following will be details on using the two models along with needed modifications, etc., to form the deep mine pollutant source model. Sample values of parameters used for the examples discussed in Chapter V, as they were listed on the computed cards, will be included in these instructions.

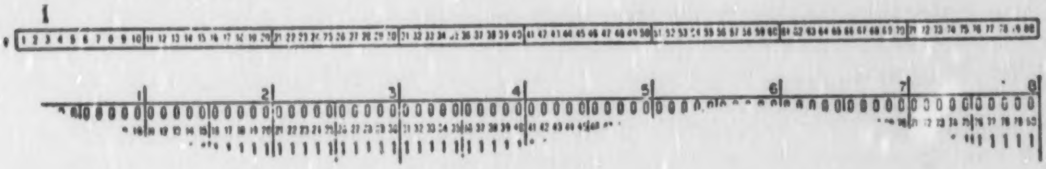
Basic Requirements

1. The computer programs for: "The Ohio State University Version of the Stanford Streamflow Simulation Model" (herein referred to as the Stanford Watershed Model - SWM) ref. 20 and "Computer Simulation of Acid Mine Drainage - AMD) ref. 22.
2. Digital Computer - this model was run on an IBM 370/165 at the Instruction and Research Computer Center (IRCC) at the Ohio State University (OSU).
3. Plotter facilities - either an IBM 1130 or IBM 1620 computer is used to drive an IBM 1627 plotter at the IRCC at OSU.

Operating Steps

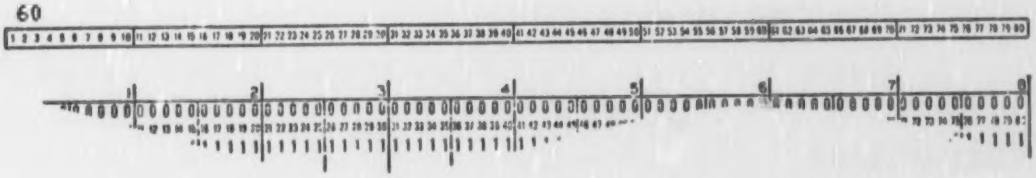
1. Part One - To generate SADD by using SWM.
   a. Determine the watershed parameters. See ref. 20 for methods of evaluating the parameters and suggested values.
   b. Transfer these values onto IBM cards. See ref. 20 for details.
   c. Run the SWM program. Class N Job at IRCC, OSU.
   d. Along with the normal SWM output optioned, punched output cards of SADD values are to be obtained (31 cards/water year). One may also obtain punched cards to plot the watershed hydrographs if this option has been requested in SWM.
2. Part Two - To calculate Minewater flow and Acid Load by AMD.
   a. Determine the parameters. These will be listed below, their definitions given, and sample values shown. Methods to determine these parameters are explained in ref. 22 and/or 8. The Sample values here are based on McDaniels' Mine, Big Four Hollow, Vinton County, Ohio.

Card #1 IMINE - number of mines being considered in the basin. This program is designed to handle up to 3 mines at a time. FORMAT (I3)



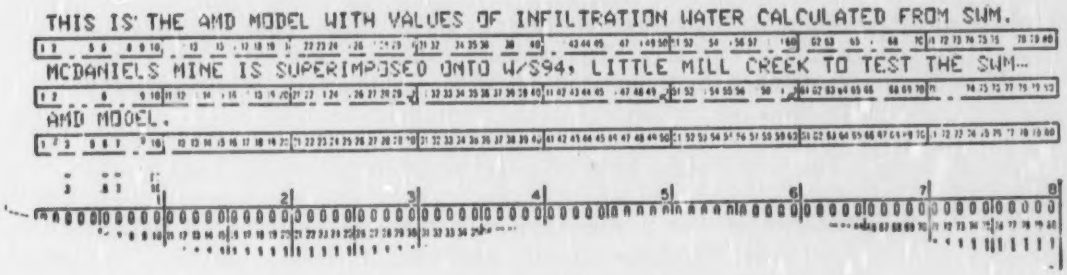
```
1
```

Card #2 N - the parameter to control the number of cards to be read in to label the title of the printed output. For example: if it takes only 1 card to label the title, N = 20; if 2 cards, N = 40, etc. FORMAT (I3)



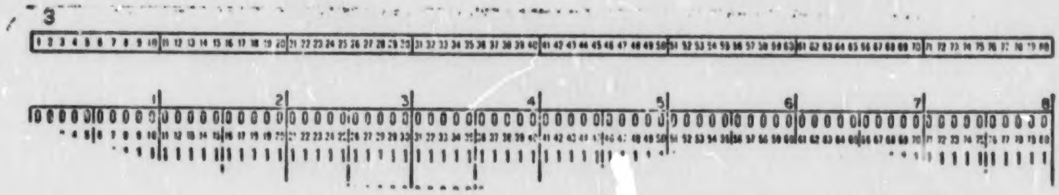
```
60
```

Cards #3 to 5 WWW - alphanumeric input to label the title of the output. Here we have 3 cards, therefore N = 60 FORMAT (20A4)



```
THIS IS THE AMD MODEL WITH VALUES OF INFILTRATION WATER CALCULATED FROM SWM.
MCDANIELS MINE IS SUPERIMPOSED ONTO W/S94, LITTLE MILL CREEK TO TEST THE SWM-
AMD MODEL.
```

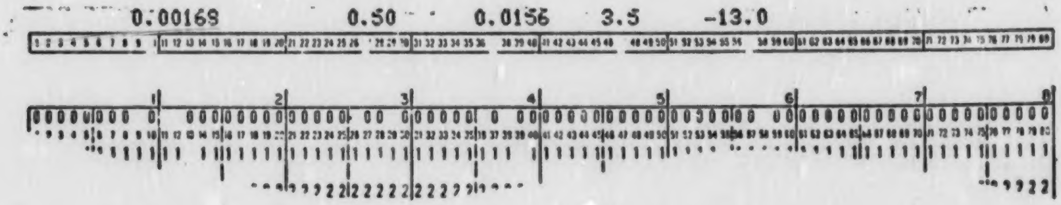
Card #6 CONTOL - program controlling device set = 3
FORMAT (I2)



Card #7 (between As and Ae, i.e., cards #7 to #61 is a DO LOOP from 1 to I mine times)

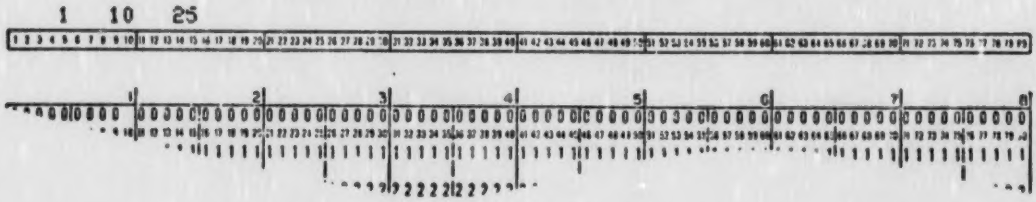
WSHED - watershed area of mine in square miles.
TANK - aquifer storage in inches.
CONFH - constant relating minewater and HEAD.
CONM - program constant see reference 22
CONB - program constant see reference 22

FORMAT (F20.10, 4F10.3)



Card #8 NFEET - number of air-solid interface increments
NLAYER - number of layers in the coal seam model
NDEPTH - number of depth increments in the model

FORMAT (3I5)



Card #9 DK - length of depth increments in feet.

ALKALI - conversion factor of acidity to alkalinity of $CaCO_3$

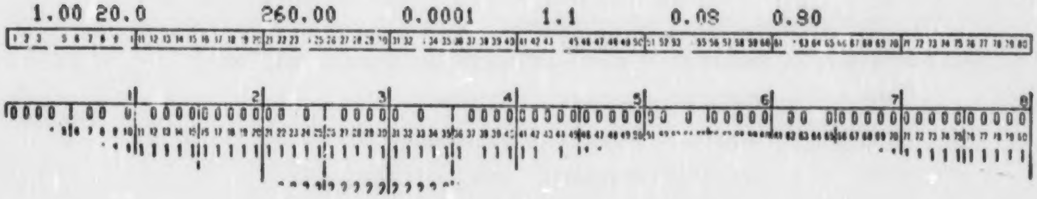
FLOWMI - minimum minewater flow rate to cause acid removal by leaching

PER - variable to determine the distance that is inundated.

WSLOPE - hypothetical slope of water level.

COND - program constant, see ref. 22

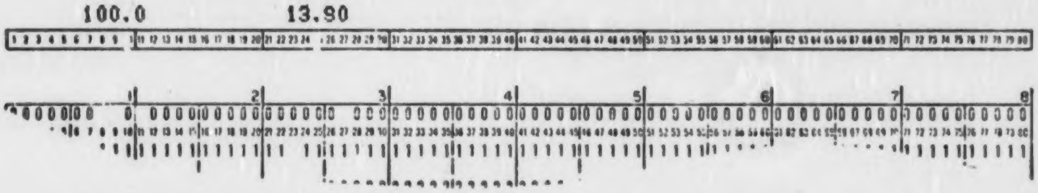
FORMAT (2F6.2, F15.3, 3F10.3, F10.5)



Card #10 DI - length of air-solid interface increments in feet.

TOP - datum plane for top of coal seam in feet

FORMAT (2F15.5)



Card #11 to Card #20 (Bs to Be, DO LOOP from 1 to NLAYER times)

ROCK TYPE - literal strata descriptors

ALT - elevation of stratum relative to datum plane in feet

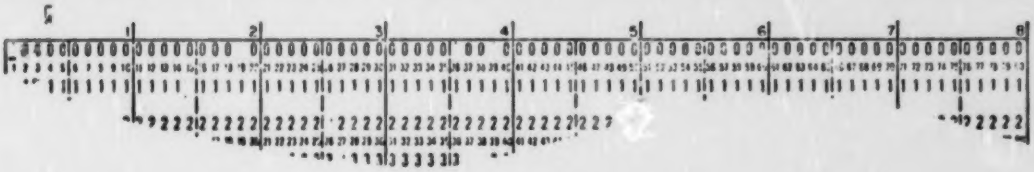
REACT - oxygen consumption rate of pyrite

PYCON - void fraction of the stratum

FORMAT (2A4, 2X, 3F10.3)

Here NLAYER = 10, therefore 10 cards

```
COAL          10.0       0.55      0.005
COAL          10.4       0.55      0.005
SHALE         10.8       2.5       0.30
COAL          10.9       0.55      0.005
COAL          11.4       0.55      0.005
COAL          11.9       0.55      0.005
COAL          12.4       0.55      0.005
SHALE         12.9       2.5       0.30
COAL          12.95      0.55      0.005
SHALE         13.40      2.5       0.30
```



Card #21 TEMP - mine temperature correction factor.

FTGMOL - volume occupied by gm moles gas

DIF - base gravity diffusion constant

CCPRFT - constant to calculate pyrite reaction rate constant

P - mine pressure

DIFF - gas diffusivity in square feet per day

GASC - gas concentration under mine conditions

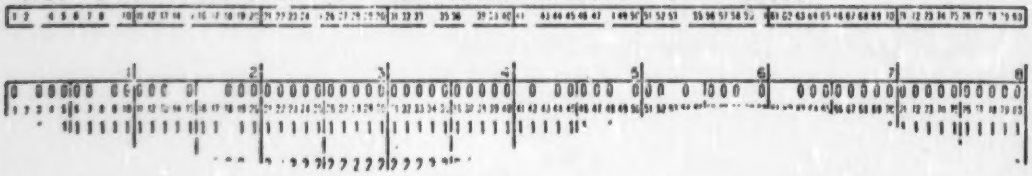
DTHETA - time increments length in days

SOX - initial total acid storage in pounds

FRACT - fraction of stored product removal by inundation

FORMAT (3FG.3, F10.3, F6.0, F6.3, 4F6.4)

```
 0.15  0.79  0.001 28287.36   454. 9.9650.08561.0    0.0    0.008
```



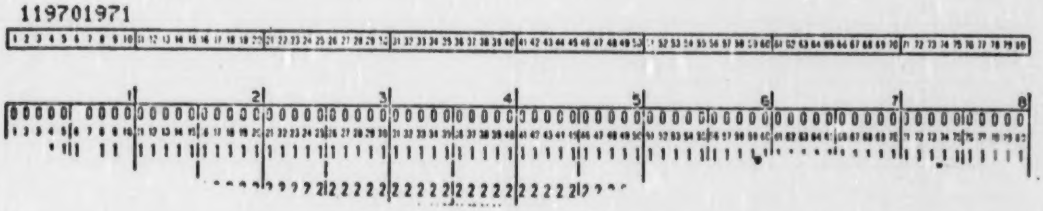
Card #22 to Card #61 (Cs to Ce, DO LOOP from 1 to NFEET, 1 to NLAYER, 1 to NDEPTH)

STORE - oxidation produce storage array. Here, NFEET = 1, NLAYER = 10, NDEPTH = 25, with 8 values of depth increment per card, and 4 cards for 25 depth increments. Therefore total cards number 1 x 10 x 4 = 40 cards.

FORMAT (8F10.2)

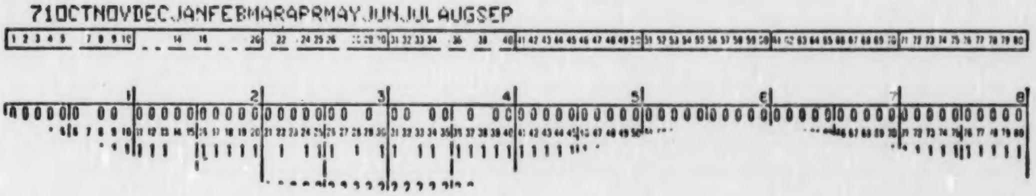
1.84	1.56	1.35	1.15	0.97	0.84	0.72	0.61
0.52	0.44	0.37	0.31	0.26	0.22	0.18	0.16
0.13	0.11	0.10	0.08	0.07	0.05	0.05	0.04
0.03							
1.66	1.43	1.24	1.08	0.94	0.83	0.71	0.62
0.54	0.46	0.39	0.32	0.26	0.22	0.19	0.17
0.15	0.13	0.11	0.10	0.08	0.07	0.06	0.05
0.04							
8.15	6.79	5.73	4.95	4.33	3.80	3.31	2.93
2.64	2.39	2.15	1.69	1.35	1.14	1.03	0.95
0.90	0.84	0.77	0.72	0.67	0.64	0.60	0.55
0.50							
2.27	2.00	1.76	1.55	1.37	1.21	1.06	0.94
0.83	0.73	0.64	0.56	0.48	0.41	0.35	0.29
0.25	0.22	0.19	0.16	0.14	0.12	0.11	0.09
0.08							
2.25	1.99	1.77	1.57	1.40	1.25	1.10	0.97
0.86	0.76	0.67	0.59	0.52	0.45	0.38	0.33
0.29	0.25	0.22	0.20	0.17	0.15	0.13	0.11
0.10							
2.18	1.94	1.72	1.53	1.36	1.21	1.07	0.95
0.85	0.76	0.67	0.60	0.53	0.47	0.40	0.34
0.29	0.26	0.23	0.20	0.18	0.16	0.14	0.13
0.11							
1.84	1.64	1.46	1.30	1.16	1.03	0.91	0.81
0.72	0.64	0.57	0.51	0.45	0.40	0.35	0.30
0.26	0.23	0.20	0.18	0.16	0.14	0.13	0.11
0.10							
4.71	4.22	3.80	3.44	3.12	2.83	2.50	2.23
2.03	1.85	1.70	1.55	1.41	1.28	1.14	0.98
0.68	0.60	0.57	0.55	0.53	0.51	0.49	0.47
0.45							
1.64	1.47	1.32	1.19	1.07	0.96	0.86	0.77
0.70	0.63	0.57	0.51	0.46	0.42	0.38	0.34
0.30	0.26	0.24	0.22	0.20	0.18	0.17	0.15
0.14							
8.86	8.48	8.12	7.78	7.46	7.16	6.84	6.55
6.28	6.04	5.80	5.58	5.36	5.16	4.96	4.76
4.50	4.27	4.10	3.97	3.85	3.73	3.61	3.50
3.39							

Card #62 IY - number of water years being studied.
YEAR 1 - beginning year of the water years
YEAR 2 - ending year of the water years
FORMAT (I2, 2I4)



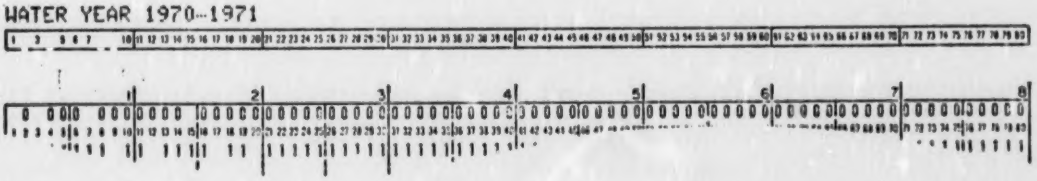
119701971

Card #63 IYR - the last two digits of YEAR 2
AMN - alphanumeric input to label the months of the water year for the plotter.
FORMAT (I4, 12A3)



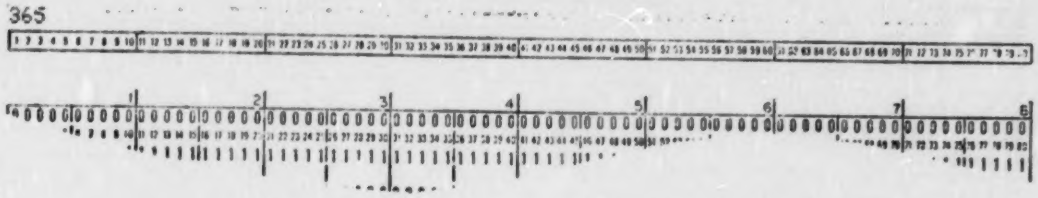
71OCTNOVDECJANFEBMARAPRMAYJUNJULAUGSEP

Card #64 C - alphanumeric input to label the water year of the plotting output.
FORMAT (6A4)

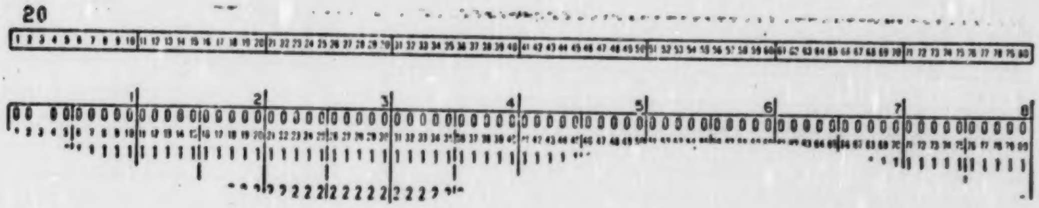


WATER YEAR 1970-1971

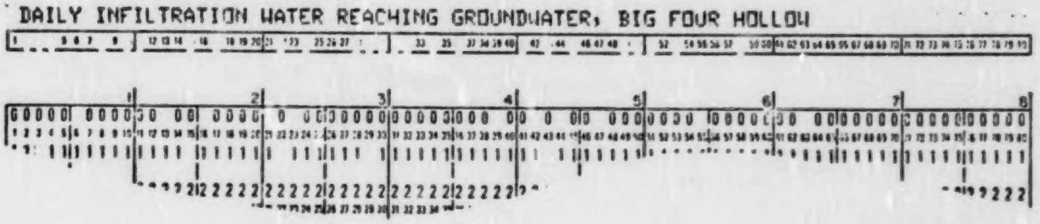
Card #65 DPY - number of days in the water year
FORMAT (I3)


365

Card #66 N same as card #2


20

Card #67 ZZZ - alphanumeric input to label output heading
FORMAT (20A4)


DAILY INFILTRATION WATER REACHING GROUNDWATER, BIG FOUR HOLLOW

Card #68 to Card #98 SADD - infiltration reaching groundwater from upper and lower zones. These data are output from the SWM

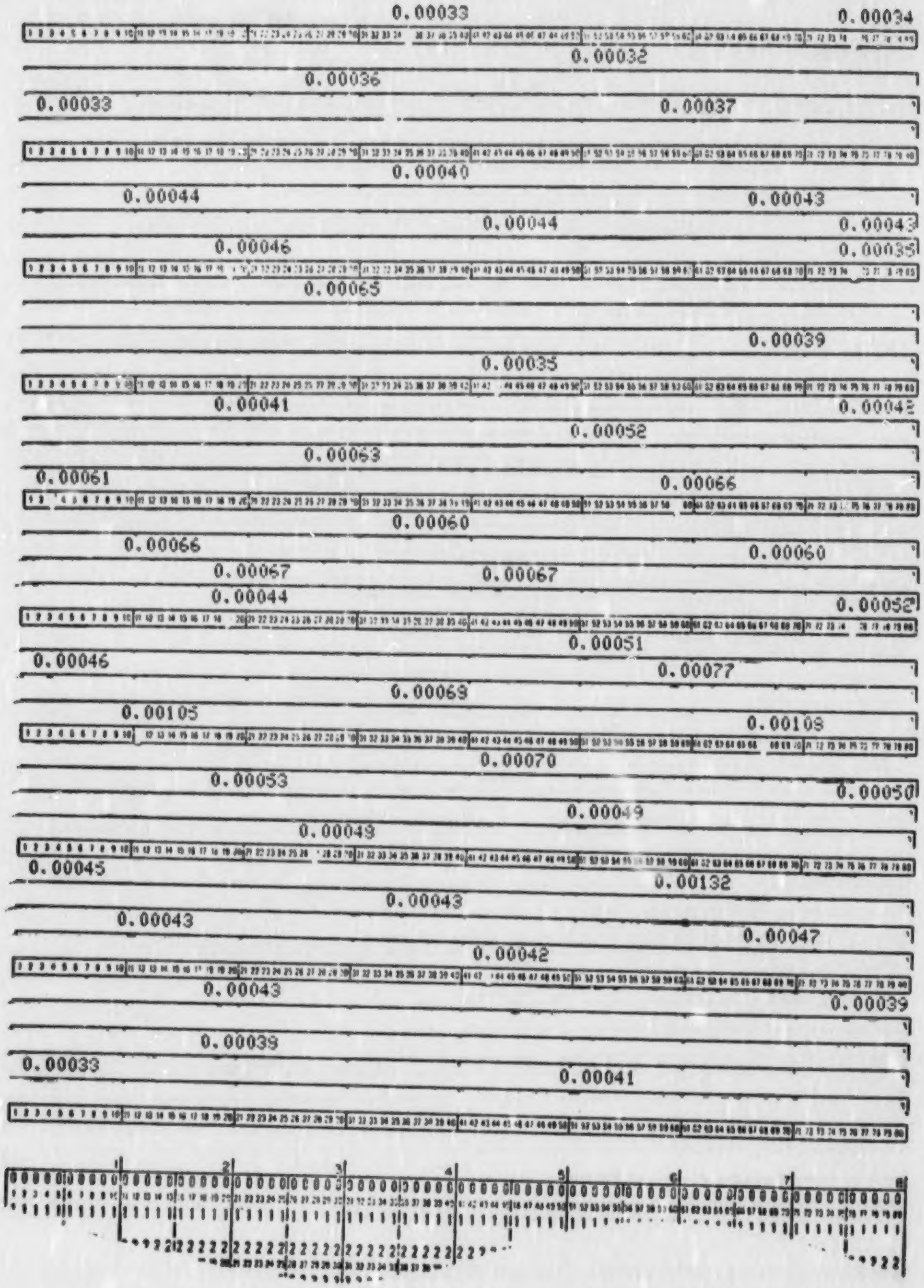
FORMAT (12F6.4)

Reproduced from best available copy.

```
0.0     0.00250.00310.00250.00340.0     0.0     0.0     0.0     0.0     0.0     0.0
0.00710.06760.00220.00200.00300.0     0.0     0.00920.0     0.0     0.0     0.0
0.00000.00399.01950.00700.00270.0     0.0     0.00000.0     0.0     0.02790.0565
0.0     0.00590.00270.05440.00240.0     0.0     0.0     0.0     0.0     0.03030.0003
0.0     0.00470.00170.00480.05430.0     0.0     0.0     0.0     0.03660.00740.0000
0.0     0.00360.00150.00250.00630.03190.0     0.09210.0     0.00030.00100.0
0.0     0.00210.00120.00210.01090.00930.0     0.15720.0     0.0     0.0     0.0
0.0     0.00060.00090.00190.05150.00350.0     0.05150.0     0.0     0.0     0.0
0.0     0.00010.00070.00150.01010.00190.0     0.00630.0     0.0     0.0     0.0
0.00630.00010.00030.00110.00500.04360.0     0.00070.0     0.11210.0     0.0
0.00250.00000.03800.00100.00420.00400.0     0.00000.0     0.05310.0     0.0
0.00010.00000.04570.00090.00340.00300.0     0.06690.0     0.00390.0     0.0059
0.00000.00000.00420.03130.01940.00170.0     0.08550.04300.00050.0     0.0057
0.17090.05170.00310.04230.00390.00060.0     0.00250.07860.00000.0     0.0000
0.00790.01500.00130.00530.00330.01260.0     0.00010.00460.02490.0     0.0
0.00460.00420.04520.00390.00290.00050.0     0.01400.00120.00010.0     0.0811
0.00160.00330.01570.00950.01570.00010.0     0.00000.00010.0     0.0     0.0009
0.00030.00270.00450.00450.00390.00000.0     0.0     0.0     0.0     0.0     0.0005
0.00010.00210.00360.00380.00790.01620.0     0.0     0.0     0.0     0.0     0.0002
0.03670.03830.00260.01350.01030.00710.0     0.0     0.0     0.0     0.0     0.0861
0.04320.00440.04810.00310.00700.00050.0     0.0     0.0     0.0     0.0     0.0024
0.00190.01340.05390.00950.06590.00000.0     0.0     0.0     0.0     0.0     0.0003
0.00110.00430.05780.00880.00800.00490.0     0.0     0.0     0.0     0.0     0.0000
0.00010.00250.00750.00630.00300.00010.0     0.0     0.0     0.00350.0     0.0
0.00000.00200.01030.00330.00210.00000.0     0.0     0.0     0.00700.10240.0
0.00000.00170.00680.01860.00110.00000.0     0.0     0.09800.00000.12040.0565
0.0     0.00130.00530.00300.00050.0     0.0     0.0     0.00190.00000.00250.0003
0.0     0.00100.00440.00270.00000.0     0.0     0.0     0.00750.0     0.00050.0004
0.09960.01670.00380.0320        0.0     0.0     0.0     0.00010.0     0.00000.0000
0.07780.00370.00330.0126        0.0     0.0     0.0     0.0     0.0     0.0     0.0
0.0108          0.000.00        0.0             0.0             0.0     0.0
```

Card #99 to Card #136 FWRECD - recorded minewater discharge in cfs.

FORMAT (10F8.5)

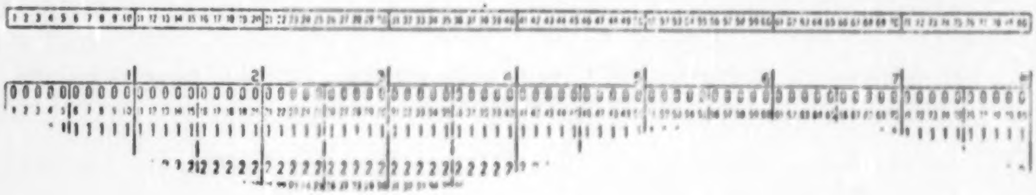

0.00033
0.00034
0.00032
0.00036
0.00033
0.00037
0.00040
0.00044
0.00043
0.00044
0.00043
0.00046
0.00035
0.00065
0.00039
0.00035
0.00041
0.00042
0.00052
0.00063
0.00061
0.00066
0.00060
0.00066
0.00060
0.00067
0.00067
0.00044
0.00052
0.00051
0.00046
0.00077
0.00069
0.00105
0.00108
0.00070
0.00053
0.00050
0.00049
0.00048
0.00045
0.00132
0.00043
0.00043
0.00047
0.00042
0.00043
0.00039
0.00039
0.00033
0.00041

Card #137 to Card #175 ACRECD - recorded acid load from mine in lb/day

FORMAT (10F8.5)

0.101 0.199
0.113
0.103
0.103 0.125
0.158
0.132 0.122
0.150 0.159
0.187 0.194
0.407
0.179
0.180
0.182 0.162
0.260
0.314
0.182 0.307
0.279
0.325 0.266
0.297 0.268 0.187
0.189
0.178
0.211 0.145
0.139 0.395
0.406
0.445 0.408
0.236
0.158 0.143
0.141
0.112
0.105 0.410
0.122
0.107 0.147
0.127
0.247 0.107
0.0945
0.109 0.0776

Card #176 blank card



b. Run program
c. Obtain printed output plus punched card output to plot on IBM 1627 the minewater flow and acid load.

3. Part Three - To plot Hydrograph (from Part 1) and Minewater Flow and Acid Load (from Part 2). See instructions on use of IBM 1627 plotter or whatever model is being used.

REFERENCES FOR APPENDIX A

1. Proceedings Fourth Symposium on Coal Mine Drainage Research. Mellon Institute, Pittsburgh. 1972.

2. Clark, C. S. Oxidation of Coal Mine Pyrite. J. Sanitary Eng. Div. ASCE. 92:127, 1966. Proc. Paper 4802.

3. Larez, A. L. Kinetics of Pyrite Oxidation. M. S. Thesis, The Ohio State University. 1970.

4. Morth, A. H. Reaction Mechanism of the Oxidation of Iron Pyrite. M. S. Thesis, The Ohio State University. 1965.

5. Smith, E. E., K. Svanks, and K. S. Shumate. Sulfide to Sulfate Reaction Studies. Proc. Second Symposium on Coal Mine Drainage Research. Pittsburgh. 1968.

6. Lau, C. H., K. S. Shumate, and E. E. Smith. The Role of Bacteria in Pyrite Oxidation Kinetics. Proc. Third Symposium on Coal Mine Drainage Research. Pittsburgh. 1970.

7. Smith, E. E. and K. S. Shumate. Pilot Scale Study of Acid Mine Drainage. Program #14010EXA, Contract 14-12-97.

8. Morth, A. H. Acid Mine Drainage: A Mathematical Model. Ph.D. Dissertation, The Ohio State University. 1971.

9. Morth, A. H., E. E. Smith, and K. S. Shumate. Pyritic Systems: A Mathematical Model. Project 14010 EAH, Office of Research and Monitoring, U. S. Environmental Protection Agency. November 1972.

10. Crawford, N. H. and R. K. Linsley. Digital Simulation in Hydrology, Stanford Model IV. Department of Civil Engineering, Stanford University. Technical Report No. 39. 1966.

11. L. D. James. Use of the Digital Computer to Analyze Hydrologic Problems. Proc. 5th Annual Sanitary and Water Resources Engineering Conference. Department of Civil Engineering, Vanderbilt University. 1966.

12. Clarke, R. D. Application of Stanford Watershed Model Concept to Predict Flood Peaks for Small Drainage Areas. Civil Engineering, Stanford University. Research Report. 1966.

13. Drooker, P. B. Application of the Stanford Watershed Model to a Small New England Watershed. M. S. Thesis. Department of Soil and Water Science, The University of New Hampshire. 1968.

14. Balk, E. L. Application of the Stanford Watershed Model to the Coshocton Hydrologic Station Data. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1968.

15. Briggs, D. L. Application of the Stanford Streamflow Simulation Model to Small Agricultural Watersheds at Coshocton, Ohio. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1968.

16. Owen, S. M. Modification of the Stanford Streamflow Watershed Model IV to Improve Groundwater Simulation for Stratified Geologic Regions. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1970.

17. Mease, W. L. A Snowmelt Subroutine for Streamflow Simulation in Ohio. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1970.

18. Valentine, L. E. Modifications of the Stanford Streamflow Simulation Model IV for Analysis of Small Watersheds. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1970.

19. Warns, J. C. User's Manual for the Ohio State University Version of the Stanford Streamflow Simulation Model IV. M. S. Thesis. Department of Civil Engineering, The Ohio State University. 1971.

20. Ricca, V. T. The Ohio State University Version of the Stanford Streamflow Simulation Model. Part I: Technical Aspects; Part II: Computer Program; Part III: User's Manual. Water Resources Center, The Ohio State University. 1972.

21. Shumate, K. S, E. E. Smith, and R. A. Brant. A Model for Pyritic Systems. (Prepared for ACS Division of the Fuel Chemistry Symposium, 157th National Meeting 13, No. 2.) 1969.

22. Chow, K. Y. Computer Simulation of Acid Mine Drainage. M. S. Thesis. Department of Chemical Engineering, The Ohio State University. 1972.

23. Kohler, M. A., T. J. Nordenson, and W. E. Fox. Evaporation from Pans and Lakes. U. S. Weather Bureau Research Paper 38. 1955.

APPENDIX B

THE REFUSE PILE AND STRIP MINE POLLUTANT SOURCE MODELS

TECHNICAL DETAILS AND COMPUTER PROGRAMS

This appendix contains details on the pollutant source model as to its history, modifications, and linking mechanisms.

An in-depth discussion of their application, showing input parameter values and selection methodology, data assembly, and typical graphical and tabular outputs is included.

The section concludes with a computer program listings for the source model.

The material presented herein was taken as much as possible from publications (papers presented and Master of Science Theses) written as part of this project during the research period.

INTRODUCTION

Today, with the shortage of some natural fuels, the coal industry may again start to grow rapidly. In the early period of America's history it was the coal industry that realized tremendous growth due to the abundance and the relative ease of obtaining the coal. Since the turn of the century through World War I the bituminous coal industry production increased from 111 million tons to 579 million tons and the number of mines increased from 2,500 to 9,300. Presently, bituminous coal production remains at about 400-600 million tons per year. Similarly, the anthracite coal industry saw a steady climb in production to a peak output of 100 million tons in 1917; in 1971 the output has declined to about 9 million tons.[7]*

---

*References listed at the end of this appendix.

The coal mining industry uses two types of mining operations to extract the coal--drift or deep mining, and strip mining. Each type of mining and coal preparation operation produces a waste pile or refuse pile of varying size, physical and mineral nature, and degree of homogeniety. Refuse piles are composed largely of shales, clays and low grade coals and often exhibit a high pyrite content. Strip mine spoil banks may be highly variable in regard to both pyrite content and location with "hot" strata and mass inclusions common. Since pyrite material near the surface is exposed to both moisture and oxygen the pyrite is chemically oxidized to produce the acid mine wastes of sulfates, ferrous iron and sulfuric acid. Significant acid production at or near the surface retards or prevents vegetative growth, and when precipitation falls on the pile, the acid materials and sediment are washed down into receiving streams where fresh pyrite material is exposed. This can lead to a continuous cycle of acid from the pile entering the nearby receiving streams. If pyrite is distributed throughout the pile, the acid load will stop only when the pile is completely washed away or when the pyrite is protected from exposure to oxygen by a natural or artificial barrier.

Today, State and Federal agencies have recognized this pollution problem and have established laws to control future mining operations. The State of Ohio has put into effect, as of April 10, 1971, the Strip Mine Act.[3] One requirement for obtaining a strip mine license, as stated in the Act, requires the operator to provide a plan that gives:

> "a description of the methods and practices the applicant intends to employ in strip mining and to prevent pollution of waters of the state, erosion, deposition of sediment, landslides, accumulation or discharge of acid water, and flooding."

The Federal government in 1969 passed the Environment Policy Act which will stop all degradation of the environment.[18] This act attacks the problem from both directions in that it calls for both future pollution prevention and for existing pollution sources to be corrected.

Acid mine drainage easily qualifies as a major pollution problem. It has been estimated that approximately 500 billion gallons per year of acid mine water containing from 5-10 million tons of sulfuric acid pollute 10,000 miles of streams and receiving waters.[12] A United States Department of Interior report indicates that surface mining operations alone seriously affect 4,800 miles of streams.[17]

The task of correcting this adverse situation can be approached in two ways: (1) by treatment of the waste or, (2) by an abatement program at the source. As in any type of pollution control, the optimum treatment/abatement alternatives for regional problems can seldom be identified through a simplistic analysis. In general, the economic, physical, and chemical interactions across an affected basin must be accounted for in greater or lesser degree. One approach would be to model a watershed of coal mining operations and then work out an optimized program considering both treatment and abatement.

In order for this type of program to work, the acid loading from the various sources must be known so that an effective program can be set up to abate or treat the acid. The deep mine pollutant source model was presented in the previous appendix. Efforts will be concentrated in this section to modeling the refuse pile and strip mine pollutant sources. The basic phenomenon for acid generation can be considered the same for both sources; differences in the hydrology and reaction kinetics do occur but these can be handled simply by modifications in the model. The reader will be informed of these differences and shown how they are modeled at the appropriate location in the narrative.

There has been limited prior work in strip mine and refuse pile models. Morth[8] has presented techniques that can be used in principle to estimate the oxidation or the acid load for either system. Sternberg and Agnew[16] have undertaken the development of a model of drainage in a surface mined area, where they were concerned with changes in ground water elevation and ground water flow that would occur in response to a uniform rate of deep percolation over the spoil bank.

In looking specifically at developing a working model for a strip mine or refuse pile there are two major areas that must be explored; how the acid is produced, and how the acid is removed.

Good's work[4] with acid production from a refuse pile at the New Kathleen Mine, near Duquoin, Illinois, comes to three general conclusions: (1) the zone of reaction extends only several inches into a pile, (2) pyrite oxidation proceeds at a relatively constant rate between rains with the acid produced accumulating in the outer mantle and, (3) only about 70% of acid salt appears in runoff, the remaining is carried into the interior of the pile later reappearing in seepage around the pile.

Brown[1] studied the transport of oxygen through layers of soil and material from the New Kathleen refuse pile. In attempting to model the transport, difficulty was encountered in estimating the diffusivity through the soil. Brown's use of zero order reaction kinetics in a refuse pile is also of questionable validity from a kinetics standpoint, although the effect on calculated pyrite oxidation rates as compared to a first order would not be great.[8]

Morth,[8] in developing his drift mine model, suggests that the same first order oxygen gradient developed for coal and shale binders could be used for a refuse pile. Morth's model did reasonably predict drift mine acid load. The main problem Morth had in utilizing his model for the other mining operation was in the hydrology of the acid mine drainage. The previous appendix showed linkage of the deep mine model with a more complicated and advanced hydrologic model than employed by Morth by using The Ohio State University version of the Stanford Streamflow Simulation Model.

The Ohio State University version of the Stanford Streamflow Simulation Model,[10] or simply the Stanford Model, is a mathematical model which synthesizes a continuous hydrograph of streamflow from climatological data and watershed parameters. The Stanford Model is designed to describe mathematically the hydrologic cycle, and Figure B.1 shows the hydrologic cycle on a refuse pile. The model keeps a chronological account of the quantities of moisture allocated to the various components of the cycle. The model is good for large watersheds, and can be applied to small watersheds down to one square mile. Thus, by knowing the various hydrological parameter needed to run the Stanford Model the streamflow hydrograph can be reasonably determined.

Therefore, if a description of the refuse pile is known, which includes the hydrological parameters and the acid production and removal parameters, a computer model can be developed that could provide reasonable predictions of acid loads in the receiving streams.

DESCRIPTION OF REFUSE PILE ACID MINE DRAINAGE MODEL

Description of a Refuse Pile

In order to write a general computer program for acid mine drainage from a refuse pile, a general sketch of a refuse pile must be given. The following outline can be used to describe most piles. A refuse pile of a coal mining operation is exactly what is says; it is a "refuse pile," or a pile made out of the material that is mined with the coal and rejected as being worthless. A pile normally consists of shale, clay and low grade coal and often exhibits a high pyrite content. Refuse piles are located near the mining operation and are associated with both deep and drift mines. The pile shape will vary with the terrain, with steep-sided piles often found in mountainous country, resulting from the dumping of material over existing steep slopes. In flat terrain, as in Illinois, the piles may be broad, and almost flat-topped, as shown in Figure B.1. On large piles there may even be ponds of water or the refuse may be used as a dam for slurry ponds if open space is available.

The typical pile (see Figure B.1) can be divided into three zones. The outer mantle, or first zone of the pile, may constitute a stratum where much of the fines (clays, powdered shales, and coal dust) have been washed out by precipitation. Thus, this zone is where the pyrite

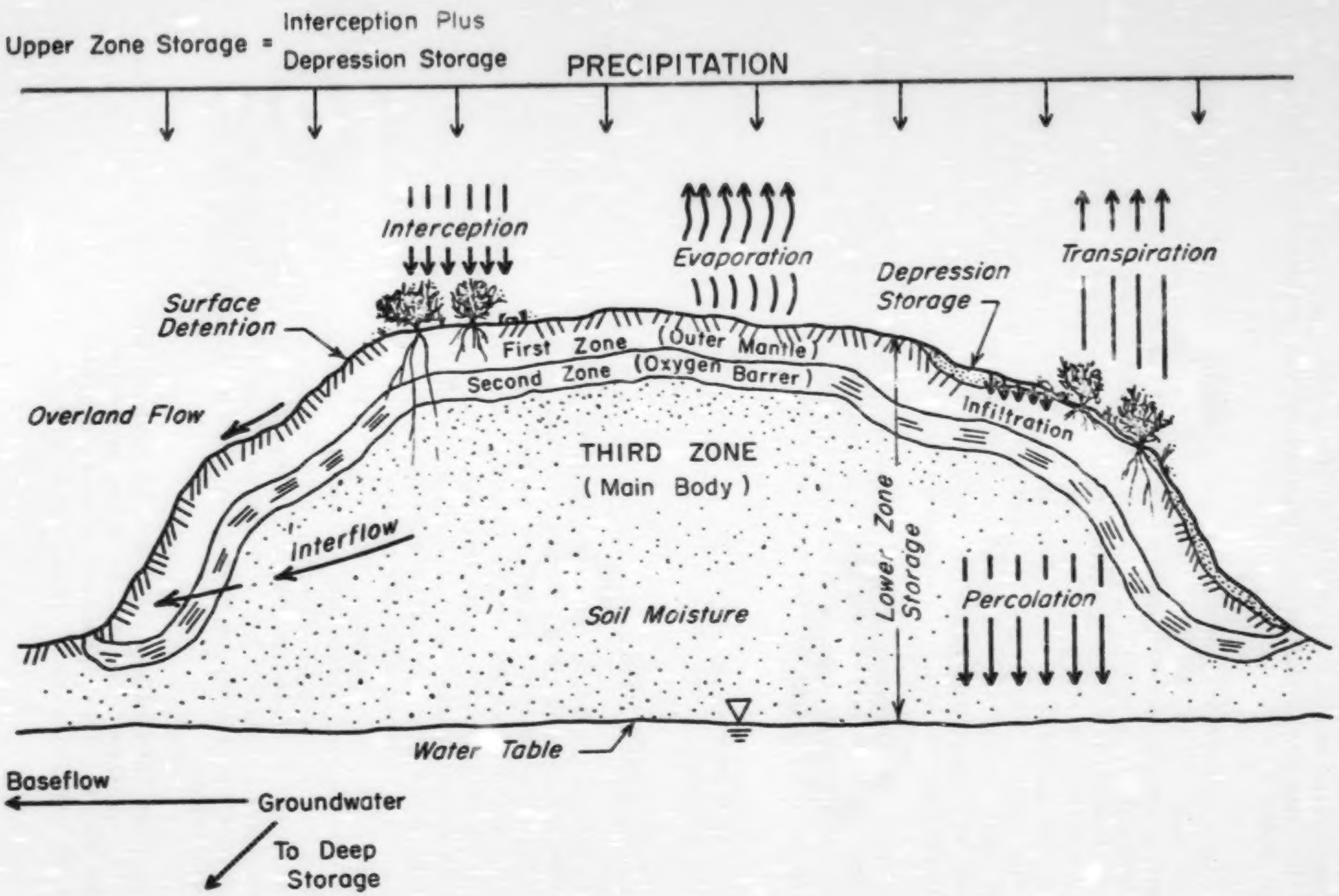

Upper Zone Storage = Interception Plus Depression Storage
PRECIPITATION
Interception
Evaporation
Transpiration
Depression Storage
Surface Detention
First Zone (Outer Mantle)
Second Zone (Oxygen Barrer)
Overland Flow
Infiltration
THIRD ZONE
(Main Body)
Interflow
Lower Zone Storage
Percolation
Soil Moisture
Water Table
Baseflow
Groundwater
To Deep Storage

Figure B.1. Hydrologic cycle on a refuse pile

oxidation occurs rapidly and where the acid products form and inhibit any vegetative cover from forming. The second zone is a layer composed of the clayey fines tightly packed by rain action into the pile and, thus, having a low permeability. The layer does have discontinuities so that water may enter the main pile but it is, on the whole, an effective water and gas barrier, preventing significant pyrite oxidation from occurring any further into the pile. The third zone is the main body of the pile and shows little evidence of weathering or pyrite oxidation.

Acid Production of a Refuse Pile

In looking at the description of a refuse pile it can be seen that the acid is produced in the outer mantle, wherein the two ingredients required for pyrite oxidation, oxygen and water, are available.

The oxidation of pyritic material can be represented by the following equations:

$$FeS_2 + 7/2\ O_2 + H_2O = Fe^{+2} + 2SO_4^{-2} + 2H^+$$

$$Fe^{+2} + 1/4\ O_2 + H^+ = Fe^{+3} + 1/2\ H_2O$$

$$Fe^{+3} + 3H_2O = Fe(OH)_3 + 3H^+$$

These equations are stochiometrically balanced, but do not define reaction mechanisms, intermediate products which may cancel out in the overall equation, or factors affecting the rate of reactions. The first equation describes the initial reactants and final products of the oxidation of pyrite by oxygen in the presence of water. The products of this initial step are the major pollutants; sulfates, ferrous iron and sulfuric acid.

Refuse Pile Model

Once the acid products have been produced in the outer mantle, water flow dictated by the hydrologic cycle is the vehicle which flushes the acid products into the stream. Now attention will be given to the hydrologic parameters needed in the actual modeling of the pile.

Hydrologic Parameters -

In looking at Figure B.1, a schematic of the hydrologic cycle, each one of the thirteen parameters will be analyzed as it applies to a refuse pile. Linsley et al.[6] provides a brief description of each parameter. When rain falls, the first part of the storm is stored on the vegetal cover as interception and in surface puddles as depression storage. As rain continues, the soil surface becomes covered with a film of water, known as surface detention, and flow begins downslope toward an established surface channel. Enroute to a channel, the water is designated as overland flow. At the same time, water is moving through the soil surface into the soil as infiltration which is distinguished from percolation, the movement of water through the soil. As the water is absorbed by the root systems of plants, where only minute portions of water remains in the plant tissues, virtually all of the water is discharged to the atmosphere as vapor through the process known as transpiration. The water known as soil moisture is near the surface where the pore space contains both air and water. Some of the water which infiltrates the soil surface may move laterally through the upper soil layer until it reaches a stream channel. This water, called interflow, moves more slowly than the surface runoff and therefore, reaches the streams somewhat later. Some precipitation may percolate downward until it reaches the water table, a theoretical line where the pore space contains only water, and move into the ground water flow. During the rain and afterwards, there is a continuous exchange of water molecules to from the atmosphere; thus, the hydrologic definitions of evaporation is the net rate of vapor transported to the atmosphere.

The objective here is to state if the parameter is needed in the refuse pile model. Transpiration is the only parameter that might be totally excluded, if the pile does not have any vegetal cover due to the acid products being produced in the outer mantle. All of the other parameters must have quantities of moisture allocated to them. It will be up to the model to see that the general water balance, inflow equals outflow plus storage, is satisfied at all times. Referring to the schematic, Figure B.1, precipitation is the inflow. The outflow consists of evaporation, transpiration, overland flow, interflow, groundwater flow, infiltration and percolation with interception, soil moisture, depression storage and surface detention as the storage. For a refuse pile the main concern is with the outflows of overland flow, interflow, and groundwater flow, since these will have the opportunity to transport the acid mine products.

Acid Production Parameters -

In keeping with developing a general model that will be applicable to many refuse piles, the pile volume within which oxidation occurs, the rate at which the oxidation occurs, and the rate of acid removal in runoff must all be described. As a first approximation, the zone of oxidation can be represented as a finite thickness outer layer of the pile, in which acid is produced at a constant rate, and from which acid is assumed to be removed at an assumed effective "saturation" concentration in the surface or subsurface runoff. For piles meeting these criteria, only the depth of outer mantle, the volumetric acid production rate, and the acid solubility are needed to describe the pile. This relatively simple model will be used for initial developments, and more complex assumptions will be discussed later.

Linking Process

In order to link the two processes it is important to know the specific amounts of four quantities: 1) acid products available, 2) precipitation falling, 3) acid products being removed, and 4) acid products remaining. Looking at Figure B.1, a schematic of the refuse pile, it can be seen as the precipitation falls on the pile, the water will saturate the area and will leave it by three main routes, overland flow, interflow, and groundwater. Initially, it will be assumed that in each case the water will become saturated with acid before it leaves. A small explanation concerning the three routes is in order. Overland flow is the water that does not enter the pile but simply runs down the outside of the pile. Once the water enters the pile it will percolate either into interflow or groundwater flow. Interflow will move through the pile laterally until it reaches the side surfaces of the pile, where it will seep out and flow overland to the stream. Because of the path interflow water must make it will take longer than overland flow to reach the stream. In order for the water to reach the groundwater flow the water must pass completely through the pile. Upon reaching the groundwater pool, the water will become baseflow to a stream or it will move to a deep storage zone if one exists in the basin.

The refuse pile model will utilize The Ohio State University version of the Stanford Streamflow Simulation Model to generate the hydrologic cycle, and to keep account of the quantities of moisture assigned to each parameter in the hydrologic cycle. The main quantities of water needed are the amount of water that is assigned to overland flow, interflow and groundwater flow. The acid in the outer mantle will be found by calculating the acid produced since the last rain and adding it to the existing acid in the mantle. Then when precipitation falls on the pile the amount of acid products that become soluble in water and leave the pile via overland flow, interflow and groundwater flow must be calculated. The precipitation waters will continuously be saturated with acid until there are no acid products left in the outer mantle to be removed. If the precipitation stops before all the acid products are removed, this remaining amount is added to the amount of acid being produced between precipitation inputs. Once precipitation falls again the cycle is restarted.

CALIBRATION OF PARAMETER COEFFICIENTS

Based on the general description previously stated, a refuse pile model can be constructed. Figure B.2 shows a step diagram of the refuse pile model. A detailed discussion of the operation of the program will occur in a following section. Attention will now focus on how the main hydrologic and acid production parameters will be formulated as needed in Step 1, 3, and 5 of the sequel below. The identification variable as used in the computer programs will be written in parentheses where

Step 1 Establish acid production characteristics of the refuse pile.
  1. Depth of outer mantle. (DEPTH)
  2. Acid production rate. (ACDPRO)
  3. Solubility of acid products. (SOLACD)
Step 2 Compute initial amount of acid available to be removed.
Step 3 Find precipitation falling on the pile and apportion the rain into 4 parts:
  1. Precipitation entering the upper zone. (ENTRUZ)
  2. Precipitation entering the lower zone. (ENTRLZ)
  3. Precipitation as interflow storage. (RGX)
  4. Precipitation as direct runoff. (VFLST)
Step 4 Find amount of acid products removed by the precipitation.
Step 5 Determine the amount of water entering the receiving stream by:
  1. Overland flow. (DIRRNF)
  2. Interflow. (INTF)
  3. Baseflow. (BASFLW)
Step 6 Calculate the amount of acid reaching the stream.
Step 7 Ascertain the amount of acid left in the pile after the rain has stopped and no flow is reaching the receiving stream.
Step 8 Return to Step 2.

Figure B.2. Step diagram of the Refuse Pile Model

applicable. A complete listing of the variables along with their units and definitions are given in Tables B.1, B.2, and B.3.

In the first step, finding the acid production* characteristics of a refuse pile, the following parameters need to be calculated; the depth of the outer mantle (DEPTH), the acid production rate (ACDPRO) and the solubility of the acid products (SOLACD). These parameters are variable and will have to be determined for each specific pile under study.

The depth of the outer mantle, the layer where the acid products are produced, can be found only by field observation. As stated earlier, the outer mantle is separated from the main pile by a second zone. The second zone, about one inch thick, is composed of clayey fines tightly packed by rain action and thus has a low permeability. In digging a hole in the pile the second zone should be easily observed and the depth of the outer mantle can be found. Where the outer mantle is of variable thickness or composition, the decrease in oxygen concentration with depth may require description in the modeling process, a refinement which will be discussed in a later section.

---

*Acid Production is defined as the net result of dynamic acid formation and product removal

Table B.1. INPUT VARIABLES OBTAINED FROM THE STANFORD WATERSHED MODEL

Variable	Units	Definition
BASFLW	in.	Current rate at which baseflow is entering channel
DAY	-	Day of month
DDYR1	-	Last two digits of first year in water year
DD23	-	Number of 15-minute periods varies from 1 to 4
DIRRNF	in.	Current rate at which direct runoff is entering channel
ENTRLZ	in.	Current rainfall entering the lower zone
ENTRUZ	in.	Current rainfall entering the upper zone
FA	-	Current month of the water year
INTF	in.	Current rate at which interflow is entering the channel
J	-	Hour of the day
OVFLST	in.	Current rainfall entering direct runoff
PR	in.	Current rainfall rate
RGX	in.	Water entering interflow storage
TOTFLW	in.	Total flow

Table B.2. INPUT VARIABLES INTO THE REFUSE PILE

Variable	Units	Definition
ACDPRO	lb acidity/acre-day/ unit depth	Acid production rate
AREA	sq. ft.	Area of refuse pile
DAY 1	-	Day to start specific output
DEPTH	feet	Depth of outer mantle
FACDEP	-	Factor for depth of outer mantle
FACDIR	-	Factor for amount of acid going into direct runoff
FACFB	-	Factor for amount of water coming from baseflow
FACFD	-	Factor for amount of water coming from direct runoff
FACFI	-	Factor for amount of water coming from interflow
FACINT	-	Factor for amount of acid going into interflow storage
FACLZ	-	Factor for amount of acid going into lower zone
FACRDS	-	Factor for amount of acid going into deep storage
FACREB	-	Factor for amount of acid going into channel by baseflow
FACRED	-	Factor for amount of acid going into channel by direct runoff
FACREI	-	Factor for amount of acid going into channel by interflow
FACUZ	-	Factor for amount of acid going into upper zone
MONTH 1	-	Month to start specific output
NDAY	-	Number of consecutive days of output requested
OPTI	-	Variable to call temperature change option
SOLACD	mg/liter	Solubility of acid products
YEAR 1	-	Year to start specific output

Table B.3. INTERNAL VARIABLES OF THE REFUSE PILE MODEL

Variable	Units	Definition
ACDDIR	lb	Amount of acid going to direct runoff
ACDINT	lb	Amount of acid going to interflow storage
ACDLZ	lb	Amount of acid going to lower zone
ACDUZ	lb	Amount of acid going to upper zone
ARB	lb	Daily acid in baseflow
ARD	lb	Daily acid in direct runoff
AREBFL	lb	Amount of acid being removed by baseflow
AREDIR	lb	Amount of acid being removed by direct runoff
AREDSR	lb	Amount of acid being removed by deep storage
AREINF	lb	Amount of acid being removed by interflow
ARI	lb	Daily acid in interflow
AMTACD	lb	Amount of acid being produced
CF	-	Conversion factor to convert inches to cubic feet
CFBAS	cfs	Flow entering channel by baseflow
CFDIR	cfs	Flow entering channel by direct runoff
CFINT	cfs	Flow entering channel by interflow
CFS	-	Conversion factor to convert inches to cubic feet per second
CFTOT	cfs	Total flow in channel
CK	-	Temperature correction factor
COUNT	-	Counter
DAYEND	-	Number of periods specific day output is requested
DDAY	-	Day
EXADIR	lb	Excess acid in direct runoff storage
EXAINT	lb	Excess acid in interflow storage
EXALZ	lb	Excess acid in lower zone
EXAUZ	lb	Excess acid in upper zone
I	-	Day
J	-	Month
SAB	lb	Monthly acid in baseflow
SAD	lb	Monthly acid in direct runoff
SAI	lb	Monthly acid in interflow
SBAS	cfs	Daily baseflow
SDRR	cfs	Daily direct runoff
SINT	cfs	Daily interflow
SLYEAR	-	Variable to see if current water year is a leap year
SSBAS	cfs	Monthly baseflow
SSDRR	cfs	Monthly direct runoff
SSINT	cfs	Monthly interflow
STACD	lb	Monthly total acid

Table B.3. CONTINUED

Variable	Units	Definition
STSTR	cfs	Monthly total flow
SUMAB	lb	Yearly acid in baseflow
SUMAD	lb	Yearly acid in direct runoff
SUMAI	lb	Yearly acid in interflow
SUMSB	cfs	Yearly baseflow
SUMSD	cfs	Yearly direct runoff
SUMSI	cfs	Yearly interflow
SUMST	cfs	Yearly total flow
SUMTA	lb	Yearly total acid load
T	°F	Current temperature
TACD	lb	Daily total acid
TIME	hr	Time interval constant (Same as used in Stanford Watershed Model)
TO	°F	Temperature at which acid production rate was determined
TOTAL	lb	Total amount of acid in outer mantle
TSTR	cfs	Daily total flow
YEARPR	-	Counter by years
YEARST	-	Beginning water year
YEARLP	-	Variable used to find what water year is a leap year

The acid production rate (ACDPRO) could be empirically calculated. If the bulk porosity of the refuse pile and the pyritic content were known, the oxygen gradient could be calculated using a first order exponential expression.[8] Another way, which was undertaken by Good,[4] presents the rate information on the basis of pounds of acid formed per day per acre of refuse pile area. Good's field data was checked by Brown's[1] laboratory rates and agreed quite well. Brown determined laboratory scale oxidation rates which can be used to estimate a rate constant for the exponential gradient calculation. At this time it is believed that the first approach would be to use Good's method in finding the pounds of acid formed per day per acre of refuse pile area. Good[4] set up a small test plot (0.109 acre) and installed a sprinkler system. Then by applying a known quantity of water and collecting all of the runoff, the acid production rate can be estimated. The second approach would be to run a laboratory study. Brown's work[1] details the approach used to obtain laboratory scale oxidation rates for refuse pile materials. The third approach would be to determine the bulk porosity (and associated oxygen diffusivity) of the refuse and the pyrite content to find the oxygen gradient as developed by Morth,[8] that would lead to an acid production rate.

The third parameter needed for the pile is the solubility of the acid products. This is an "effective" solubility, which reflects the varying availability of acid to the runoff components on and within the pile, and will reach, as a maximum value, the true solubility of the acid products. In general, the effective solubility will be less than true solubility, and must be determined by trial and error, with field water quality data as a rough guide.

Step 3 deals with the initial precipitation, that must be assigned to four zones in the pile. In the program this data will be input data generated from a hydrologic model. In this study, The Ohio State University version of the Stanford Streamflow Simulation Model will be employed. The Stanford Watershed Model will also provide the amounts of water reaching the streams, which will be needed in Step 5.

The Stanford Watershed Model is a complex program that keeps a chronological account of the quantities of moisture allocated to the various components of the hydrologic cycle. The model simulates the actual hydrologic condition of the watershed and works best with watersheds of 100 acres or greater.

In order to understand how the Stanford Watershed Model computes the parameters needed in Steps 3 and 5, a brief description will be given of each parameter. The variable notation as used in the Stanford Watershed Model will be retained. For a detailed explanation of the parameters refer to the Stanford Watershed Model.[10]

The current rainfall rate (PR) is the current amount of precipitation entering the pile. The Stanford Model includes a snowmelt subroutine. In order to utilize this snowmelt option, the basic operation of the subroutine will have to be recalibrated, as the snow on refuse piles normally melts sooner than on the surrounding terrain.[9] If the snowmelt subroutine is used in the Stanford Model, then the precipitation falling may be in the form of snow. The snow may not enter the pile immediately, but will enter later when the snow melts by additional rainfall, radiation, conduction, convection or condensation. Thus in the winter months, November to March, the precipitation is checked to see if it is rain or snow and then by considering the various snowmelt mechanisms the current rainfall rate entering the pile is found. This rate is based on hourly data, but can be subdivided into fractions of an hour by linear approximation.

The current rain entering the upper zone (ENTRUZ) is equal to the current rainfall rate (PR) minus the residual rainfall after soil surface moisture depletion (P4). P4, the residual rainfall after soil surface moisture depletion is found by finding the residual rainfall after interception multiplied by the fraction of incoming moisture that is not retained in the upper zone storage. The above calculation is based on several factors; current interception rate, maximum interception rate for a dry watershed, soil surface moisture index, lake evaporation, daily pan evaporation and monthly pan evaporation coefficient.

The current precipitation entering the lower zone (ENTRLZ) is equal to the residual rainfall after soil surface moisture depletion (P4) minus the sum of the current moisture entering surface runoff plus interflow (SHRD). This latter sum equals the square of residual rainfall after soil surface moisture depletion (P4) divided by twice the current peak infiltration rate (D4F). The two quantities, D4F and P4, depend on evaporation, infiltration index, soil moisture index, interception and moisture not retained in the upper zone.

The current direct runoff (OVFLST or RX) equals the sum of current moisture entering surface runoff plus interflow (SHRD) divided by a variable controlling entry of moisture into interflow (C3). SHRD has already been defined. The variable controlling entry of moisture into interflow (C3) is dependent on the interflow index and the current ratio of soil moisture storage to the soil moisture storage index.

The current water entering interflow storage (RGX) equals the sum of current moisture entering surface runoff plus interflow (SHRD) minus the current direct runoff (RX). These quantities have been previously described.

A check to see that the current rainfall rate equals the sum of its four parts can be made as follows:

Current rainfall rate (PR) = current rain entering the upper zone (ENTRUZ) + current rain entering the lower zone (ENTRLZ) + current direct runoff (RX) + current water entering interflow storage (RGX)

or

PR = (ENTRUZ = PR - P4) + (ENTRLZ = P4 - SHRO) + (RX = RX) + (RGX = SHRO - RX).

Therefore, PR = PR.

After the initial precipitation is calculated, the Stanford Watershed Model can then route the water through the pile or watershed; it simulates the total stream flow and its component parts, overland flow, interflow, and baseflow.

The current rate at which overland flow enters the stream is based on turbulent range equations.[10] The Chezy-Manning equation was used to derive a relationship between surface detention storage at equilibrium, the supply rate to overland flow, Manning's n and the length and slope of the flow plane. An empirical relationship developed by Crawford and Linsley between outflow depth and detention storage for reproducing experimental hydrographs was used. By combining the above equations a rate of discharge from overland flow can be found.

Thus the Stanford Watershed Model simulates overland flow by continuously solving the continuity equations, by setting the surface detention at the end of the time interval equal to the surface detention at the end of the previous time interval plus the increment added to surface detention during the time interval, minus losses.

The current rate at which interflow is entering the channel or stream (INTF) is modeled by a logarithmic decay equation, $S_t = -q_t/\ell n K_r$, where S_t is the storage at time t, q_t is the flow at time t, and $\ell n K_r$ is the natural logarithm of the interflow recession constant.

The baseflow (BASFLW) is equal to the ground water baseflow (GWF). The ground water baseflow (GWF) is computed by a logarithmic decay equation. The equation is the same as for the stream's interflow, with a modification which permits increased groundwater flow to reflect changes in the recession constant due to wet antecedent conditions.

A basic description of the data needed for the Stanford Watershed Model has been presented; a brief description of the input parameters needed to run the Watershed Model now will be given. For detailed calculations and formulas see reference (10).

DATA NEEDED	DERIVED INPUT PARAMETERS
Topographic Map or Aerial Photographs	1. Time of concentration and time area histograms
	2. Watershed drainage area
	3. Impervious fraction of watershed surface
	4. Estimate of stream and lake surface areas
	5. Mean overland flow path length
	6. Average ground slope of overland flow surface perpendicular to the channel
Soil Borings or Observation Wells	1. Soil type
	2. Soil porosity
	3. Soil specific yield
	4. Soils permeability
	5. Groundwater fluctuation

Climatologic Data	1. Daily dewpoint temperature
	2. Daily wind movement
	3. Daily solar radiation
	4. Maximum and minimum daily temperature
	5. Daily lake and pan evaporation
	6. Hourly recorded rainfall
	7. Storage-gage daily rainfall
Streamflow Data	1. Daily streamflow records
	2. Daily diverted flows
Physical Inspection of the Area	1. Watershed cover
	2. Swamps and extensive soil cracks
	3. Manning's roughness for overland flow on soil surface
	4. Manning's roughness on impervious surface
	5. Manning's roughness value for stream surface
	6. Capacity of channel.

There are additional parameters that must be determined by a trial and error approach. Therefore, for optimum calibration results, 3-5 years of climatologic and streamflow data are required to permit the Watershed Model to stabilize its soil moisture condition.

MODEL TESTING

Based upon the descriptions presented earlier, a computer program to model a refuse pile was written. The program is very general in nature, the intent being that it will be applicable to various refuse piles. In developing the program three assumptions were made.

1. The Ohio State University version of the Stanford Streamflow Simulation Model, models a watershed in which the refuse pile or piles are located. It is also possible to have the total watershed as a refuse pile.

2. The acid is produced at a uniform rate and is stopped when precipitation is falling because the outer mantle will become saturated and insufficient amounts of oxygen will be present for oxidation of the pyritic material.

3. The solubility concentration of the acid products is constant, and the total volume of water will always be saturated.

Tables B.1, B.2, and B.3 list the nomenclature as used in the computer program, i.e., the input variables obtained from the Stanford Watershed Model, the input variables into the Refuse Pile Model, and the internal variables in the Refuse Pile Model.

A step diagram was presented earlier (Figure B.2); at this time a detailed explanation will be given to the computations in each step using the actual computer language to explain all equations. The computer variables are listed in Tables B.1-B.3 and the program statement listing is included at the end of this presentation of the Refuse Pile Model.

Since the acid production rate varies with a change in temperature, a temperature option is available which will reflect the acid production rate change as seen in equations (B.1) and (B.2).

$$\text{CK} = 2.**((\text{T} - \text{TO})/18.) \qquad \text{(B.1)}$$

Unit analysis:

None

$$\text{ACDPRO} = \text{ACDPRO} * \text{CK} \qquad \text{(B.2)}$$

Unit analysis:

$$\frac{\text{lb}}{\text{ACRE - DAY} \quad \text{UNIT DEPTH(FT)}} = \frac{\text{lb}}{\text{ACRE - DAY} \quad \text{UNIT DEPTH(FT)}}$$

The change will double the acid production rate for every 10°C temperature change.[12] Note: The above equation uses fahrenheit units.

Computing the initial amount of acid available to be removed was done in Step 2. This amounts to keeping track of the amount of acid in the outer mantle. When no precipitation is falling the acid in the outer mantle is produced at a uniform rate, but if precipitation is falling no acid products are being formed because the oxygen required for oxidation is assumed to be absent and some of the products are being washed out of the mantle. The program will check the input and if there is no rain, the amount of acid formed is given by equation (B.3).

$$\text{AMTACD} = \frac{\text{ACDPRO* TIME* AREA* DEPTH* FACDEP}}{\text{24* 43500}} \qquad \text{(B.3)}$$

Unit analysis:

$$\text{lb} = \frac{\frac{\text{lb}}{\text{ACRE, DAY, UNIT DEPTH(FT)}} * \text{HRS} * \text{FT}^2 * \text{FT} * \text{None}}{\frac{\text{HR}}{\text{DAY}} * \frac{\text{FT}^2}{\text{ACRE}}}$$

If it is raining, some of the acid products are washed out of the outer mantle and the amount of acid remaining is found by equation B.4.

$$\text{TOTAL} = \text{AMTACD} - \text{ACDINT} - \text{ACDLZ} - \text{ACDUZ} - \text{ACDDIR} \qquad \text{(B.4)}$$

Unit analysis:

$$\frac{\text{lb}}{\text{Time Interval}} = \frac{\text{lb}}{\text{Time Interval}} - \frac{\text{lb}}{\text{Time Interval}} - \frac{\text{lb}}{\text{Time Interval}} - \frac{\text{lb}}{\text{Time Interval}} - \frac{\text{lb}}{\text{Time Interval}}$$

In Step 3, finding the amount of precipitation falling on the pile and apportioning the rain into four parts, upper zone, lower zone, interflow storage, and direct runoff, is done by the Stanford Watershed Model. Refer to earlier dicussions for details.

Once the precipitation starts falling the water will become saturated with the acid products. These products will move into the same four areas along with the precipitation. Step 4 finds this amount by the following equations:

$$\text{ACDDIR} = \text{FACDIR} * \text{SOLACD} * \text{OVFLST} * \text{CF} \qquad \text{(B.5)}$$

$$\text{ACDINT} = \text{FACINT} * \text{SOLACD} * \text{RGX} * \text{CF} \qquad \text{(B.6)}$$

$$\text{ACDLZ} = \text{FACLZ} * \text{SOLACD} * \text{ENTRLZ} * \text{CF} \qquad \text{(B.7)}$$

$$\text{ALDUZ} = \text{FALUZ} * \text{SOLACD} * \text{ENTRUZ} * \text{CF} \qquad \text{(B.8)}$$

Unit analysis:

$$\frac{\text{lb}}{\text{Time Interval}} = \text{None} * \frac{\text{lb}}{\text{FT}^3} * \frac{\text{in}}{\text{Time Interval}} * \frac{\text{FT}^3}{\text{in}}$$

The above four equations ascertain the amount of acid removed by the rain during one time interval. In order to keep track of the acid in each zone at all times, the products removed must be added together, as accomplished by equations (B.9)-(B.12). These four equations show how the computer updates the amount of acid in each zone after each time interval; the variables are defined in Tables B.1-B.3.

$$\mathrm{EXADIR} = \mathrm{ACDDIR} + \mathrm{EXADIR} \tag{B.9}$$

$$\mathrm{EXAINT} = \mathrm{ACDINT} + \mathrm{EXAINT} \tag{B.10}$$

$$\mathrm{EXALZ} = \mathrm{ACDLZ} + \mathrm{EXALZ} \tag{B.11}$$

$$\mathrm{EXAUZ} = \mathrm{ALDUZ} + \mathrm{EXAUZ} \tag{B.12}$$

Unit analysis:

$$\frac{\text{lb}}{\text{Time Interval}} = \frac{\text{lb}}{\text{Time Interval}} + \frac{\text{lb}}{\text{Time Interval}}$$

In Step 5, the amount of water entering the receiving stream; by overland flow, interflow, and baseflow, was determined. This was explained in detail earlier.

By knowing how much water reaches the receiving stream, it is possible to calculate the amount of acid being carried to the stream by the three routes of overland flow, interflow, and baseflow, as done in Step 6. The equations used are:

$$\mathrm{AREDIR} = \mathrm{DIRRNF} * \mathrm{SOLALD} * \mathrm{FACRED} * \mathrm{CF} \tag{B.13}$$

$$\mathrm{AREINF} = \mathrm{INTF} * \mathrm{SOLACO} * \mathrm{FACREI} * \mathrm{CF} \tag{B.14}$$

$$\mathrm{AREBFL} = \mathrm{BASFLW} * \mathrm{SOLACO} * \mathrm{FACREB} * \mathrm{CF} \tag{B.15}$$

Unit analysis:

$$\frac{\text{lb}}{\text{Time Interval}} = \frac{\text{in.}}{\text{Time Interval}} * \frac{\text{lb}}{\text{FT}^3} * \text{None} * \frac{\text{FT}^3}{\text{in.}}$$

The computer variables are defined in Tables B.1, B.2, and B.5. The above equations would determine the amount of acid products that would reach the stream providing there were sufficient products available throughout the runoff period; if not the quantities would become zero. The acid products being conveyed in the overland flow would come from storage on the surface and in the direct runoff until it is depleted.

Next the acid products would be supplied by the acid stored in the upper zone. Acid products being conveyed by interflow would be controlled by the amount of acid retained in interflow storage and then from the acid accumulated in the lower zone. Baseflow can only obtain the acid products from the lower zone. There are also acid products given to the groundwater which may not appear as baseflow but will go to deep storage. This acid is obtained from the lower zone.

In ascertaining the amount of acid products left in the pile, done in Step 7, the products leaving the pile to go to the receiving stream or deep storage are subtracted from the acidic products brought to the four areas in Step 4. The following equations show how this continual updating was done

$$\text{EXADIR} = \text{EXADIR} - \text{AREDIR} \qquad \text{(first this)} \qquad (\text{B.16})$$

$$\text{EXAUZ} = \text{EXAUZ} - \text{AREDIR} \qquad \text{(then this)} \qquad (\text{B.17})$$

$$\text{EXAINT} = \text{EXAINT} - \text{AREINF} \qquad \text{(first this)} \qquad (\text{B.18})$$

$$\text{EXALZ} = \text{EXALZ} - \text{AREINF} \qquad \text{(then this)} \qquad (\text{B.19})$$

$$\text{EXALZ} = \text{EXALZ} - \text{AREBFL} \qquad \text{(always)} \qquad (\text{B.20})$$

$$\text{EXALZ} = \text{EXALZ} - \text{AREDSR} \qquad \text{(always)} \qquad (\text{B.21})$$

Unit analysis:

$$\frac{\text{lb}}{\text{Time Interval}} = \frac{\text{lb}}{\text{Time Interval}} - \frac{\text{lb}}{\text{Time Interval}}$$

Tables B.1-B.3 define the computer variables. The Refuse Pile computer program (near the end of this Appendix) is listed for the operators described in Steps 1 through 8 of Figure B.2. Prior to the program listing are instructions detailing the necessary changes needed in the Stanford Watershed Model to generate the information required by the Refuse Pile Model.

Presently, the program outputs various tables plus specific day(s) information. The following Tables B.4-B.7 show the standard output items for the daily acid load in direct runoff, interflow, baseflow and the total acid load. Tables B.8-B.10 gives the daily flow reaching the receiving stream by direct runoff, interflow and baseflow, and the total flow reaching the stream is presented in Table B.11. Monthly summaries of acid load and flows are given in Table B.12; Table B.13 shows specific day output. The values listed in the tables have no specific

Table B.4. DAILY ACID LOAD IN DIRECT RUNOFF

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.0	0.0	0.0	35.04	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	576.86	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	6.97	0.0	0.30	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	2234.01	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	12498.32	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	68.83	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	2.32	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	2.84	6.32	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	9.03	0.0	36.22	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	142.48	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	37035.92	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.61	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	8.30	0.0	0.0	0.0
27	0.0	0.0	18.32	0.0	0.0	0.0	65.35	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	757.91	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	********	0.0	5.29	5.80	0.0	0.0	0.0	0.0
30	0.0	0.0	0.0	5.72	********	0.0	4.69	0.0	0.0	0.0	0.0	0.0
31	0.0	********	0.0	0.0	********	0.0	********	0.0	********	0.0	0.0	********

Table B.5. DAILY ACID LOAD IN INTERFLOW

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.0	0.0	0.0	11525.72	24.76	1079.42	16.94	651.53	83.80	0.0	0.0	0.04
2	0.0	0.0	139.04	1155.39	24.76	628.75	12779.44	24.76	26.87	0.0	0.0	0.0
3	0.0	0.0	112.17	24.76	1430.74	42.99	1309.74	24.76	0.0	0.0	0.0	0.0
4	0.0	0.0	390.39	24.76	1655.42	0.0	58.64	24.76	0.0	0.0	21.24	0.0
5	0.0	0.0	1005.21	24.76	123.78	189.95	24.76	24.76	0.0	0.0	24.76	0.0
6	0.0	0.0	102.97	24.76	24.76	205.51	24.76	24.76	0.0	0.0	24.76	0.0
7	0.0	0.0	24.76	24.76	24.76	16.51	24.76	24.76	0.0	0.0	20.64	0.0
8	0.0	0.0	24.76	24.76	24.76	16.51	24.76	24.76	0.0	0.0	0.0	0.0
9	0.0	0.0	24.76	24.76	10560.96	676.43	20.64	24.76	0.0	0.0	0.0	0.0
10	0.0	0.0	24.76	24.76	41393.79	92.52	0.0	17.28	0.0	0.0	0.0	0.0
11	0.0	0.0	24.76	13.41	1174.91	24.76	0.0	16.51	0.0	0.0	0.0	0.0
12	0.0	0.0	24.76	0.0	24.76	24.76	0.0	856.62	4826.77	0.0	0.0	0.0
13	0.0	0.0	24.76	0.0	1701.34	24.76	0.0	19.69	4136.71	0.0	0.0	0.0
14	0.0	0.0	24.76	1464.92	2022.99	2657.29	0.0	0.0	31.13	0.0	0.0	0.0
15	0.0	0.47	24.76	7051.47	666.06	11634.86	0.0	0.0	24.76	0.0	0.0	0.0
16	0.0	0.0	30.61	733.61	24.76	446.15	0.0	0.0	24.76	0.0	0.0	0.0
17	0.0	0.0	10.36	24.76	24.76	24.76	0.0	0.0	24.76	0.0	0.0	0.0
18	0.0	0.0	8.60	24.76	24.76	12.38	0.0	0.0	24.76	0.0	0.0	0.0
19	0.0	0.0	82.42	9.29	24.76	0.0	0.0	236.49	24.76	0.0	0.0	0.0
20	0.0	0.0	25.84	5985.09	24.76	0.0	3.57	24.76	24.76	0.0	0.0	0.0
21	0.0	0.0	24.76	80054.87	156.20	0.0	4.13	27.56	20.64	0.0	0.0	0.0
22	0.0	0.0	223.10	5163.65	276.07	0.0	4.13	24.76	2815.42	0.0	0.0	0.0
23	0.0	0.0	509.31	34.27	1885.28	0.0	4.13	4.13	336.08	0.0	0.0	0.0
24	0.0	0.0	226.58	24.76	200.36	0.0	4.13	0.0	24.76	0.09	0.0	0.0
25	0.0	0.0	63.37	24.76	24.76	0.0	4.13	0.0	244.30	0.0	0.0	0.0
26	0.0	0.0	457.93	24.76	24.76	1.20	15.44	4.73	3146.36	0.0	0.0	0.0
27	0.0	0.0	2552.98	24.76	24.76	3.22	11302.31	60.67	636.27	0.0	0.0	0.0
28	0.0	0.0	125.50	24.76	11.35	6.25	36864.66	22.83	24.76	0.0	0.0	0.0
29	0.0	0.0	24.76	652.27	********	8.25	2666.27	1302.14	20.64	0.0	0.0	0.0
30	0.0	0.0	24.76	3996.66	********	8.25	3236.04	773.68	0.0	0.0	0.0	0.69
31	0.0	********	24.76	265.92	********	8.25	********	24.76	********	0.0	0.0	********

Table B.6. DAILY ACID LOAD IN BASEFLOW

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.64	6.84	67.07	734.69	1444.79	975.11	507.16	1793.39	679.83	1702.07	178.86	28.72
2	0.0	33.02	143.43	1010.37	1338.16	1077.79	651.79	1657.01	693.24	1596.30	165.27	45.40
3	0.0	33.02	324.18	943.47	1248.22	1043.73	704.06	1530.95	640.62	1474.89	152.72	44.20
4	0.0	29.06	576.81	876.40	1317.01	965.14	713.36	1414.69	592.63	1362.76	198.64	41.27
5	0.0	28.38	770.29	812.42	1383.05	906.15	659.68	1307.38	547.57	1259.91	203.62	37.15
6	0.0	24.76	866.60	752.06	1289.84	964.11	609.63	1207.97	505.96	1163.60	188.14	34.74
7	0.0	12.64	827.39	695.65	1193.18	922.32	563.57	1116.31	467.95	1075.21	174.04	33.02
8	0.0	6.28	782.67	643.88	1103.24	852.15	521.27	1031.87	432.70	993.17	160.97	28.89
9	75.84	25.28	737.44	595.04	1106.16	851.46	481.54	953.62	399.85	917.84	148.76	28.20
10	0.0	32.50	692.04	549.98	1667.50	883.97	445.42	899.27	369.75	848.02	137.41	26.83
11	0.0	28.89	647.67	508.54	1708.78	816.83	411.71	842.52	341.72	784.22	127.78	28.72
12	0.0	27.69	603.64	470.01	1590.45	754.95	380.41	1162.06	454.19	729.70	123.48	24.76
13	0.0	24.76	561.51	434.42	1570.33	697.71	351.52	1242.03	1316.15	674.49	114.37	24.76
14	0.0	24.08	521.78	429.94	1537.31	855.24	325.04	1148.99	1224.14	623.25	105.59	20.98
15	0.0	28.89	483.43	682.23	1573.26	1490.19	300.45	1061.96	1131.96	575.95	97.51	20.64
16	0.0	35.43	452.99	881.04	1461.81	1570.51	277.92	981.30	1045.97	532.27	90.12	19.26
17	12.30	33.02	423.73	818.44	1353.30	1451.32	257.11	907.18	966.69	492.20	83.75	16.51
18	5.68	31.64	420.31	759.11	1250.62	1340.91	237.50	836.05	893.42	455.57	76.87	16.51
19	0.0	34.57	434.41	704.25	1155.52	1240.31	219.44	845.44	825.84	424.09	71.54	15.99
20	0.0	33.02	435.97	787.49	1067.98	1144.51	222.37	790.92	763.24	409.31	65.87	12.38
21	0.0	28.89	407.42	1663.89	1011.23	1057.49	212.57	754.29	705.62	378.35	60.88	12.38
22	0.0	28.20	416.02	1946.11	1002.63	977.35	196.40	699.26	1128.18	349.80	56.41	12.38
23	0.0	24.76	559.10	1822.63	1081.05	903.23	181.95	649.39	1641.71	323.66	52.11	12.38
24	0.0	24.59	670.02	1696.74	1182.18	835.12	168.02	612.93	1520.98	310.08	48.33	8.94
25	10.32	20.64	695.65	1573.60	1092.92	771.32	155.30	566.67	1480.91	298.55	44.89	8.25
26	15.48	25.11	686.54	1456.14	1009.85	717.32	144.46	539.66	1802.85	276.02	41.27	8.25
27	2.58	28.38	705.45	1346.93	933.15	690.66	725.06	551.70	2320.33	255.39	39.38	8.25
28	7.48	24.76	690.14	1244.61	872.44	695.13	1376.52	563.57	2145.95	235.78	37.15	8.25
29	0.0	24.76	646.12	1234.11	********	642.17	1650.48	564.43	1982.22	218.07	33.71	8.25
30	0.0	22.01	601.75	1478.67	********	593.49	1721.33	754.64	1832.94	206.72	31.99	32.85
31	0.0	********	558.93	1556.75	********	548.76	********	699.69	********	193.30	28.89	********

Table B.7. DAILY ACID LOAD

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED TOTAL ACID LOAD IN POUNDS

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.64	6.84	67.07	12295.44	1469.55	2054.54	524.10	2444.93	763.62	1702.07	178.86	28.76
2	0.0	33.02	282.47	2165.77	1362.93	1706.54	14008.09	1681.78	720.11	1596.30	165.27	45.40
3	0.0	33.02	436.35	968.24	2685.92	1086.73	2014.13	1555.71	640.62	1474.89	152.72	44.20
4	0.0	29.06	967.21	901.16	2972.44	965.14	772.01	1439.46	592.63	1362.76	219.87	41.27
5	0.0	28.38	1775.51	837.19	1506.83	1096.10	684.64	1332.14	547.57	1259.91	228.39	37.15
6	0.0	24.76	969.56	776.82	1314.60	1169.62	634.60	1232.74	505.96	1163.60	212.91	34.74
7	0.0	12.64	852.15	720.41	1217.95	938.83	588.33	1141.07	467.95	1075.21	194.68	33.02
8	0.0	6.28	807.43	668.65	1128.00	868.66	546.03	1056.63	432.70	993.17	160.97	28.89
9	75.84	25.28	762.20	619.81	13901.14	1527.90	502.17	978.38	399.85	917.84	148.76	28.20
10	0.0	32.50	716.80	574.75	55559.64	976.49	445.42	916.56	369.75	848.02	137.41	26.83
11	0.0	28.89	672.43	521.95	2883.69	841.66	411.71	859.03	341.72	784.22	127.78	28.72
12	0.0	27.69	628.41	470.01	1615.22	779.75	380.41	2018.68	5349.79	729.70	123.48	24.76
13	0.0	24.76	586.27	434.42	3274.00	722.48	351.52	1261.72	5452.79	674.49	114.37	24.76
14	0.0	24.08	546.54	1897.70	3566.63	3512.54	325.04	1148.99	1255.27	623.25	105.59	20.98
15	0.0	29.37	508.19	7742.75	2239.33	13163.27	300.45	1061.96	1156.72	575.95	97.51	20.64
16	0.0	35.43	483.60	1614.66	1486.58	2016.66	277.92	981.30	1070.73	532.27	90.12	19.26
17	12.30	33.02	444.09	843.21	1378.06	1476.09	257.11	907.18	991.45	492.20	83.75	16.51
18	5.68	31.64	428.91	783.87	1275.39	1353.30	237.50	838.05	918.19	455.57	76.87	16.51
19	0.0	34.57	516.84	713.53	1180.29	1240.31	219.44	1083.94	850.60	424.09	71.54	15.99
20	0.0	33.02	461.80	6915.05	1092.75	1144.51	225.94	815.69	788.00	409.31	65.87	12.38
21	0.0	28.89	432.18	*********	1167.43	1057.49	216.69	781.85	726.26	378.35	60.88	12.38
22	0.0	28.20	639.11	7109.73	1278.70	977.35	200.53	724.03	3951.22	349.80	56.41	12.38
23	0.0	24.76	1068.42	1856.69	2966.33	903.23	186.08	653.51	1977.79	323.66	52.11	12.38
24	0.0	24.59	896.61	1721.50	1382.53	835.12	172.15	612.93	1545.74	310.16	48.33	8.94
25	10.32	20.64	759.02	1398.37	1117.69	771.32	159.42	566.67	1725.20	298.55	44.89	8.25
26	15.48	25.11	1144.47	1480.90	1034.62	718.52	159.90	544.40	4957.49	276.02	41.27	8.25
27	2.58	26.38	3276.76	1371.70	957.92	693.89	12092.70	612.37	2956.61	255.39	39.38	8.25
28	7.48	24.76	815.65	1269.37	883.79	703.39	38999.06	586.40	2170.71	235.78	37.15	8.25
29	0.0	24.76	670.88	1886.39	*********	650.42	4322.04	1872.38	2002.86	218.07	33.71	8.25
30	0.0	22.01	626.51	5481.01	*********	601.75	4962.05	1528.33	1832.94	206.72	31.99	33.54
31	0.0	*********	563.69	1822.67	*********	557.03	*********	723.85	*********	193.30	28.89	*********

Table B.8. DAILY DIRECT RUNOFF REACHING RECEIVING STREAM

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECOND

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.0	0.0	0.0	0.1247	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	2.0528	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0248	0.0	0.0011	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0	0.0	7.9499	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0	0.0	44.4763	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2450	0.0	0.0	0.0
13	0.0	0.0	0.0	0.0	0.0083	0.0	0.0	0.0	0.0	0.0	0.0	0.0
14	0.0	0.0	0.0	0.0101	0.0225	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	0.0	0.0	0.0	0.0321	0.0	0.1360	0.0	0.0	0.0	0.0	0.0	0.0
16	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
17	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
18	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
20	0.0	0.0	0.0	0.5070	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
21	0.0	0.0	0.0	131.7553	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0271	0.0	0.0	0.0
23	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
25	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0295	0.0	0.0	0.0
27	0.0	0.0	0.0652	0.0	0.0	0.0	0.2326	0.0	0.0	0.0	0.0	0.0
28	0.0	0.0	0.0	0.0	0.0	0.0	2.6971	0.0	0.0	0.0	0.0	0.0
29	0.0	0.0	0.0	0.0	********	0.0	0.0188	0.0207	0.0	0.0	0.0	0.0
30	0.0	0.0	0.0	0.0203	********	0.0	0.0167	0.0	0.0	0.0	0.0	0.0
31	0.0	********	0.0	0.0	********	0.0	********	0.0	********	0.0	0.0	********

Table B.9. DAILY INTERFLOW REACHING RECEIVING STREAM

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	0.0	0.0	0.0	41.0152	0.0881	3.8412	0.0603	2.3185	0.2982	0.0	0.0	0.0002
2	0.0	0.0	0.4948	4.1115	0.0881	2.2375	45.4767	0.0881	0.0956	0.0	0.0	0.0
3	0.0	0.0	0.3992	0.0881	5.0914	0.1530	4.6608	0.0881	0.0	0.0	0.0	0.0
4	0.0	0.0	1.3892	0.0881	5.8909	0.0	0.2087	0.0881	0.0	0.0	0.0756	0.0
5	0.0	0.0	3.5771	0.0881	0.4405	0.6760	0.0881	0.0881	0.0	0.0	0.0881	0.0
6	0.0	0.0	0.3664	0.0881	0.0881	0.7313	0.0881	0.0881	0.0	0.0	0.0881	0.0
7	0.0	0.0	0.0881	0.0881	0.0881	0.0588	0.0881	0.0881	0.0	0.0	0.0734	0.0
8	0.0	0.0	0.0881	0.0881	0.0881	0.0588	0.0881	0.0881	0.0	0.0	0.0	0.0
9	0.0	0.0	0.0881	0.0881	37.5821	2.4071	0.0734	0.0881	0.0	0.0	0.0	0.0
10	0.0	0.0	0.0881	0.0881	147.3032	0.3293	0.0	0.0615	0.0	0.0	0.0	0.0
11	0.0	0.0	0.0881	0.0477	4.1810	0.0881	0.0	0.0588	0.0	0.0	0.0	0.0
12	0.0	0.0	0.0881	0.0	0.0881	0.0881	0.0	3.0484	17.1765	0.0	0.0	0.0
13	0.0	0.0	0.0881	0.0	6.0544	0.0881	0.0	0.0701	14.7209	0.0	0.0	0.0
14	0.0	0.0	0.0881	5.2130	7.1990	9.4562	0.0	0.0	0.1108	0.0	0.0	0.0
15	0.0	0.0017	0.0881	25.0932	2.3702	41.4035	0.0	0.0	0.0881	0.0	0.0	0.0
16	0.0	0.0	0.1089	2.6106	0.0881	1.5877	0.0	0.0	0.0881	0.0	0.0	0.0
17	0.0	0.0	0.0369	0.0881	0.0881	0.0881	0.0	0.0	0.0881	0.0	0.0	0.0
18	0.0	0.0	0.0306	0.0881	0.0881	0.0441	0.0	0.0	0.0881	0.0	0.0	0.0
19	0.0	0.0	0.2933	0.0330	0.0881	0.0	0.0	0.8467	0.0881	0.0	0.0	0.0
20	0.0	0.0	0.0920	21.2984	0.0881	0.0	0.0127	0.0881	0.0881	0.0	0.0	0.0
21	0.0	0.0	0.0881	284.8818	0.5559	0.0	0.0147	0.0981	0.0734	0.0	0.0	0.0
22	0.0	0.0	0.7939	18.3752	0.9824	0.0	0.0147	0.0881	10.0189	0.0	0.0	0.0
23	0.0	0.0	1.8124	0.1219	6.7089	0.0	0.0147	0.0147	1.1960	0.0	0.0	0.0
24	0.0	0.0	0.8063	0.0881	0.7130	0.0	0.0147	0.0	0.0881	0.0003	0.0	0.0
25	0.0	0.0	0.2255	0.0881	0.0881	0.0	0.0147	0.0	0.8693	0.0	0.0	0.0
26	0.0	0.0	1.6296	0.0881	0.0881	0.0043	0.0549	0.0168	11.1966	0.0	0.0	0.0
27	0.0	0.0	9.0850	0.0881	0.0881	0.0115	40.2201	0.2159	2.2642	0.0	0.0	0.0
28	0.0	0.0	0.4466	0.0881	0.0404	0.0294	131.1858	0.0812	0.0881	0.0	0.0	0.0
29	0.0	0.0	0.0881	2.3212	*********	0.0294	9.4881	4.6338	0.0734	0.0	0.0	0.0
30	0.0	0.0	0.0881	14.2225	*********	0.0294	11.5157	2.7532	0.0	0.0	0.0	0.0024
31	0.0	*********	0.0881	0.9463	*********	0.0294	*********	0.0881	*********	0.0	0.0	*********

Table B.10. DAILY BASEFLOW REACHING RECEIVING STREAM

ANNUAL SUMMARY FCR WATER YEAR 1958 - 1959
SYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	2.4307	0.1095	0.2387	2.6144	5.1414	3.4700	1.8048	6.3819	2.4192	6.0570	0.6365	0.1022
2	1.0667	0.1175	0.5104	3.5955	4.7620	3.8354	2.3195	5.8966	2.4670	5.6806	0.5881	0.1616
3	0.9865	0.1175	1.1536	3.3574	4.4419	3.7142	2.5055	5.4480	2.2797	5.2485	0.5435	0.1573
4	0.9119	0.1034	2.0526	3.1187	4.6867	3.4345	2.5386	5.0343	2.1089	4.8495	0.7069	0.1469
5	0.8439	0.1010	2.7411	2.8911	4.9217	3.2246	2.3482	4.6524	1.9486	4.4835	0.7246	0.1322
6	0.7803	0.0881	3.0839	2.6763	4.5900	3.4309	2.1701	4.2987	1.8005	4.1408	0.6695	0.1236
7	0.7215	0.0881	2.9443	2.4755	4.2460	3.2821	2.0055	3.9725	1.6652	3.8262	0.6193	0.1175
8	0.6677	0.0759	2.7852	2.2913	3.9260	3.0324	1.8550	3.6720	1.5398	3.5343	0.5728	0.1028
9	0.6218	0.0900	2.6242	2.1175	3.9364	3.0300	1.7136	3.3935	1.4229	3.2662	0.5294	0.1004
10	0.5826	0.1157	2.4627	1.9572	5.9339	3.1457	1.5851	3.2001	1.3156	3.0178	0.4890	0.0955
11	0.5392	0.1026	2.3048	1.8097	6.0808	2.9070	1.4651	2.9982	1.2160	2.7907	0.4547	0.1022
12	0.4988	0.0985	2.1481	1.6726	5.6596	2.6867	1.3537	4.1353	1.6163	2.5967	0.4394	0.0881
13	0.4608	0.0881	1.9982	1.5459	5.5682	2.4629	1.2509	4.4198	4.6836	2.4002	0.4070	0.0881
14	0.4266	0.0857	1.6568	1.5300	5.4706	3.0435	1.1567	4.0888	4.3562	2.2179	0.3758	0.0747
15	0.3941	0.1028	1.7203	2.4278	5.5986	5.3030	1.0692	3.7791	4.0282	2.0496	0.3470	0.0734
16	0.3648	0.1261	1.6120	3.1353	5.2020	5.5688	0.9690	3.4921	3.7222	1.8941	0.3207	0.0685
17	0.3372	0.1175	1.5435	2.9125	4.8158	5.1647	0.9149	3.2283	3.4400	1.7515	0.2980	0.0588
18	0.3152	0.1126	1.4957	2.7014	4.4504	4.7717	0.8452	2.9823	3.1793	1.6212	0.2736	0.0588
19	0.2901	0.1230	1.5459	2.5061	4.1120	4.4137	0.7809	3.0086	2.9388	1.5092	0.2546	0.0569
20	0.2699	0.1175	1.5514	2.8023	3.6005	4.0728	0.7913	2.8146	2.7160	1.4565	0.2344	0.0441
21	0.2485	0.1028	1.4498	5.9211	3.5985	3.7632	0.7564	2.6842	2.5110	1.3464	0.2166	0.0441
22	0.2295	0.1004	1.4804	6.9254	3.5679	3.4780	0.6969	2.4884	4.0147	1.2448	0.2007	0.0441
23	0.2130	0.0881	1.9896	6.4860	3.8470	3.2142	0.6475	2.2109	5.8421	1.1518	0.1854	0.0441
24	0.1971	0.0875	2.3643	6.0380	4.2069	2.9718	0.5979	2.1812	5.4125	1.1034	0.1720	0.0318
25	0.1818	0.0734	2.4755	5.5998	3.8892	2.7448	0.5526	2.0165	5.2699	1.0624	0.1597	0.0294
26	0.1726	0.0894	2.4431	5.1818	3.5936	2.5526	0.5141	1.9204	6.4156	0.9823	0.1469	0.0294
27	0.1603	0.1010	2.5104	4.7932	3.3207	2.4578	2.5602	1.9633	6.2571	0.9088	0.1401	0.0294
28	0.1475	0.0881	2.4559	4.4290	3.1047	2.4737	4.8984	2.0055	7.6365	0.8391	0.1322	0.0294
29	0.1365	0.0881	2.2993	4.3917	*********	2.2652	5.8733	2.0086	7.0539	0.7760	0.1200	0.0294
30	0.1285	0.0783	2.1414	5.2620	*********	2.1120	6.1255	2.6854	6.5227	0.7356	0.1138	0.1169
31	0.1175	*********	1.9890	5.5398	*********	1.9529	*********	2.4878	*********	0.6879	0.1028	*********

Table B.11. DAILY FLOW REACHING RECEIVING STREAM

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959
SYNTHESIZED STREAMFLOW IN CUBIC FEET PER SECOND

DAY	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP
1	2.4307	0.1095	0.2387	43.7543	5.2295	7.3113	1.8651	8.7005	2.7174	6.0570	0.6365	0.1024
2	1.0667	0.1175	1.0052	7.7071	4.8501	6.0729	49.8490	5.9847	2.5626	5.6806	0.5881	0.1616
3	0.9865	0.1175	1.5528	3.4455	9.5581	3.8672	7.1674	5.5361	2.2797	5.2465	0.5435	0.1573
4	0.9119	0.1034	3.4419	3.2069	10.5777	3.4345	2.7472	5.1224	2.1089	4.8495	0.7824	0.1469
5	0.8439	0.1010	6.3183	2.9792	5.3622	3.9006	2.4364	4.7405	1.9486	4.4835	0.8127	0.1322
6	0.7803	0.0881	3.4503	2.7644	4.6781	4.1622	2.2583	4.3868	1.8005	4.1408	0.7577	0.1236
7	0.7215	0.0881	3.0324	2.5637	4.3342	3.3409	2.0936	4.0606	1.6652	3.8262	0.6928	0.1175
8	0.6677	0.0759	2.8733	2.3794	4.0141	3.0912	1.9431	3.7601	1.5398	3.5343	0.5728	0.1028
9	0.6218	0.0900	2.7124	2.2056	49.4683	5.4372	1.7870	3.4816	1.4229	3.2662	0.5294	0.1004
10	0.5826	0.1157	2.5508	2.0453	197.7135	3.4749	1.5851	3.2616	1.3158	3.0178	0.4890	0.0955
11	0.5392	0.1028	2.3929	1.8574	10.2619	2.9951	1.4651	3.0569	1.2160	2.7907	0.4547	0.1022
12	0.4988	0.0985	2.2362	1.6726	5.7479	2.7748	1.3537	7.1836	19.0377	2.5967	0.4394	0.0881
13	0.4608	0.0881	2.0863	1.5459	11.6506	2.5710	1.2509	4.4899	19.4042	2.4002	0.4070	0.0881
14	0.4266	0.0857	1.9449	6.7531	12.6921	12.4997	1.1567	4.0688	4.4670	2.2179	0.3758	0.0747
15	0.3941	0.1045	1.8084	27.5531	7.9688	46.8425	1.0692	3.7791	4.1163	2.0496	0.3470	0.0734
16	0.3648	0.1261	1.7209	5.7459	5.2901	7.1765	0.9690	3.4921	3.8103	1.8941	0.3207	0.0685
17	0.3372	0.1175	1.5603	3.0006	4.9039	5.2528	0.9149	3.2283	3.5282	1.7515	0.2980	0.0568
18	0.3152	0.1126	1.5263	2.7895	4.5386	4.8158	0.8452	2.9823	3.2674	1.6212	0.2736	0.0588
19	0.2901	0.1230	1.8392	2.5392	4.2001	4.4137	0.7809	3.8573	3.0269	1.5092	0.2546	0.0569
20	0.2699	0.1175	1.6434	24.6078	3.8886	4.0728	0.8040	2.9027	2.8042	1.4565	0.2344	0.0441
21	0.2485	0.1028	1.5379	422.5950	4.1544	3.7632	0.7711	2.7823	2.5845	1.3464	0.2166	0.0441
22	0.2295	0.1004	2.2743	25.3006	4.5504	3.4780	0.7136	2.5765	14.0607	1.2448	0.2007	0.0441
23	[illegible]	0.0881	3.8020	6.6079	10.5559	3.2142	0.6622	2.3256	7.0381	1.1518	0.1854	0.0441
24	0.1971	0.0875	3.1907	6.1261	4.9198	2.9718	0.6126	2.1812	5.5006	1.1037	0.1720	0.0318
25	0.1818	0.0734	2.7010	5.6879	3.9774	2.7448	0.5673	2.0165	6.1393	1.0624	0.1597	0.0294
26	0.1726	0.0894	4.0727	5.2699	3.6818	2.5569	0.5690	1.9373	17.6416	0.9823	0.1469	0.0294
27	0.1603	0.1010	11.6606	4.8813	3.4088	2.4652	43.0328	2.1792	10.5213	0.9088	0.1401	0.0294
28	0.1475	0.0881	2.9025	4.5172	3.1451	2.5031	138.7814	2.0868	7.7247	0.8391	0.1322	0.0294
29	0.1365	0.0881	2.3874	6.7129	*********	2.3146	15.3603	6.6630	7.1273	0.7760	0.1200	0.0294
30	0.1285	0.0783	2.2295	19.5047	*********	2.1414	17.6579	5.4387	6.5227	0.7356	0.1138	0.1193
31	0.1175	*********	2.0771	6.4661	*********	1.9823	*********	2.5759	*********	0.6879	0.1028	*********

Table B.12. MONTHLY SUMMARY OF ACID LOADS AND FLOWS

ANNUAL SUMMARY FOR WATER YEAR 1958 - 1959

OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	YEAR TOTAL
SYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS												
0.	0.	18.	37231.	14748.	38.	1410.	6.	85.	0.	0.	0.	53536.
SYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS												
0.	0.	6388.	118478.	63606.	17836.	68393.	4268.	16523.	0.	91.	1.	295585.
SYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS												
130.	786.	17420.	31110.	35546.	29237.	15373.	29688.	30855.	21140.	3140.	669.	215095.
THESIZED TOTAL ACID LOAD IN POUNDS												
130.	786.	23827.	186817.	113899.	47111.	85177.	33963.	47463.	21141.	3232.	670.	564216.
SYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECOND												
0.	0.	0.	132.	52.	0.	5.	0.	0.	0.	0.	0.	191.
SYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND												
0.	0.	23.	422.	226.	63.	243.	15.	59.	0.	0.	0.	1052.
SYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND												
15.	3.	62.	111.	126.	104.	55.	106.	110.	75.	11.	2.	781.
SYNTHESIZED TOTAL AMOUNT OF WATER ENTERING THE STREAM IN CUBIC FEET PER SECOND												
15.	3.	85.	665.	405.	168.	303.	121.	169.	75.	12.	2.	2023.

Table B.13. SPECIFIC DAY OUTPUT

```
 YR MO DY HR PD   AREDIR     AREINF     AREBFL     ARETOT    CFDIR      CFINT      CFBAS      CFTOT
 58  4 20  1  1   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  1  2   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  1  3   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  1  4   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  2  1   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  2  2   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  2  3   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  2  4   0.0        0.0        7.0512     7.0512    0.0        0.0        0.0251     0.0251
 58  4 20  3  1   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  3  2   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  3  3   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  3  4   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  4  1   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  4  2   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  4  3   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  4  4   0.0        0.0        7.0082     7.0082    0.0        0.0        0.0249     0.0249
 58  4 20  5  1   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  5  2   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  5  3   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  5  4   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  6  1   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  6  2   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  6  3   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  6  4   0.0        0.0        6.9652     6.9652    0.0        0.0        0.0248     0.0248
 58  4 20  7  1   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  7  2   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  7  3   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  7  4   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  8  1   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  8  2   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  8  3   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  8  4   0.0        0.0        6.9222     6.9222    0.0        0.0        0.0246     0.0246
 58  4 20  9  1   0.0        0.1290     6.9222     7.0512    0.0        0.0005     0.0246     0.0251
 58  4 20  9  2   0.0        0.3670     6.9222     7.3091    0.0        0.0014     0.0246     0.0260
 58  4 20  9  3   0.0        0.7309     6.9222     7.6531    0.0        0.0026     0.0246     0.0272
 58  4 20  9  4   0.0        1.0319     6.9222     7.9540    0.0        0.0037     0.0246     0.0283
 58  4 20 10  1   0.0        1.5478     7.0512     8.5990    0.0        0.0055     0.0251     0.0306
 58  4 20 10  2   0.0        2.0638     7.0512     9.1149    0.0        0.0073     0.0251     0.0324
 58  4 20 10  3   0.0        2.4937     7.0512     9.5449    0.0        0.0089     0.0251     0.0340
 58  4 20 10  4   0.0        2.9236     7.0512     9.9748    0.0        0.0104     0.0251     0.0355
 58  4 20 11  1   0.0        3.6116     7.2231    10.8347    0.0        0.0129     0.0257     0.0386
 58  4 20 11  2   0.0        4.2135     7.2231    11.4366    0.0        0.0150     0.0257     0.0407
 58  4 20 11  3   0.0        4.8154     7.2231    12.0386    0.0        0.0171     0.0257     0.0428
 58  4 20 11  4   0.0        5.3744     7.2231    12.5975    0.0        0.0191     0.0257     0.0448
 58  4 20 12  1   0.0        6.1913     7.4381    13.6294    0.0        0.0220     0.0265     0.0485
 58  4 20 12  2   0.0        6.9652     7.4381    14.4033    0.0        0.0248     0.0265     0.0513
 58  4 20 12  3   0.0        7.6961     7.4381    15.1342    0.0        0.0274     0.0265     0.0539
 58  4 20 12  4   0.0        8.3840     7.4381    15.8221    0.0        0.0298     0.0265     0.0563
 58  4 20 13  1   0.0        9.3729     7.6961    17.0689    0.0        0.0334     0.0274     0.0607
 58  4 20 13  2   0.0       10.3618     7.6961    18.0578    0.0        0.0369     0.0274     0.0643
 58  4 20 13  3   0.0       11.2646     7.6961    18.9607    0.0        0.0401     0.0274     0.0675
 58  4 20 13  4   0.0       12.0816     7.6961    19.7776    0.0        0.0430     0.0274     0.0704
 58  4 20 14  1   0.0       13.3284     7.9970    21.3254    0.0        0.0474     0.0285     0.0759
 58  4 20 14  2   0.0       14.4893     7.9970    22.4863    0.0        0.0516     0.0285     0.0800
 58  4 20 14  3   0.0       15.5641     7.9970    23.5612    0.0        0.0554     0.0285     0.0838
 58  4 20 14  4   0.0       16.5960     7.9970    24.5930    0.0        0.0591     0.0285     0.0875
 58  4 20 15  1   0.0       17.4559     8.2980    25.7539    0.0        0.0621     0.0295     0.0916
 58  4 20 15  2   0.0       18.2728     8.2980    26.5706    0.0        0.0650     0.0295     0.0946
 58  4 20 15  3   0.0       19.0037     8.2980    27.3017    0.0        0.0676     0.0295     0.0972
 58  4 20 15  4   0.0       19.7346     8.2980    28.0326    0.0        0.0702     0.0295     0.0998
 58  4 20 16  1   0.0       20.2936     8.5990    28.8925    0.0        0.0722     0.0306     0.1028
 58  4 20 16  2   0.0       20.8525     8.5990    29.4515    0.0        0.0742     0.0306     0.1048
 58  4 20 16  3   0.0       21.3254     8.5990    29.9244    0.0        0.0759     0.0306     0.1065
```

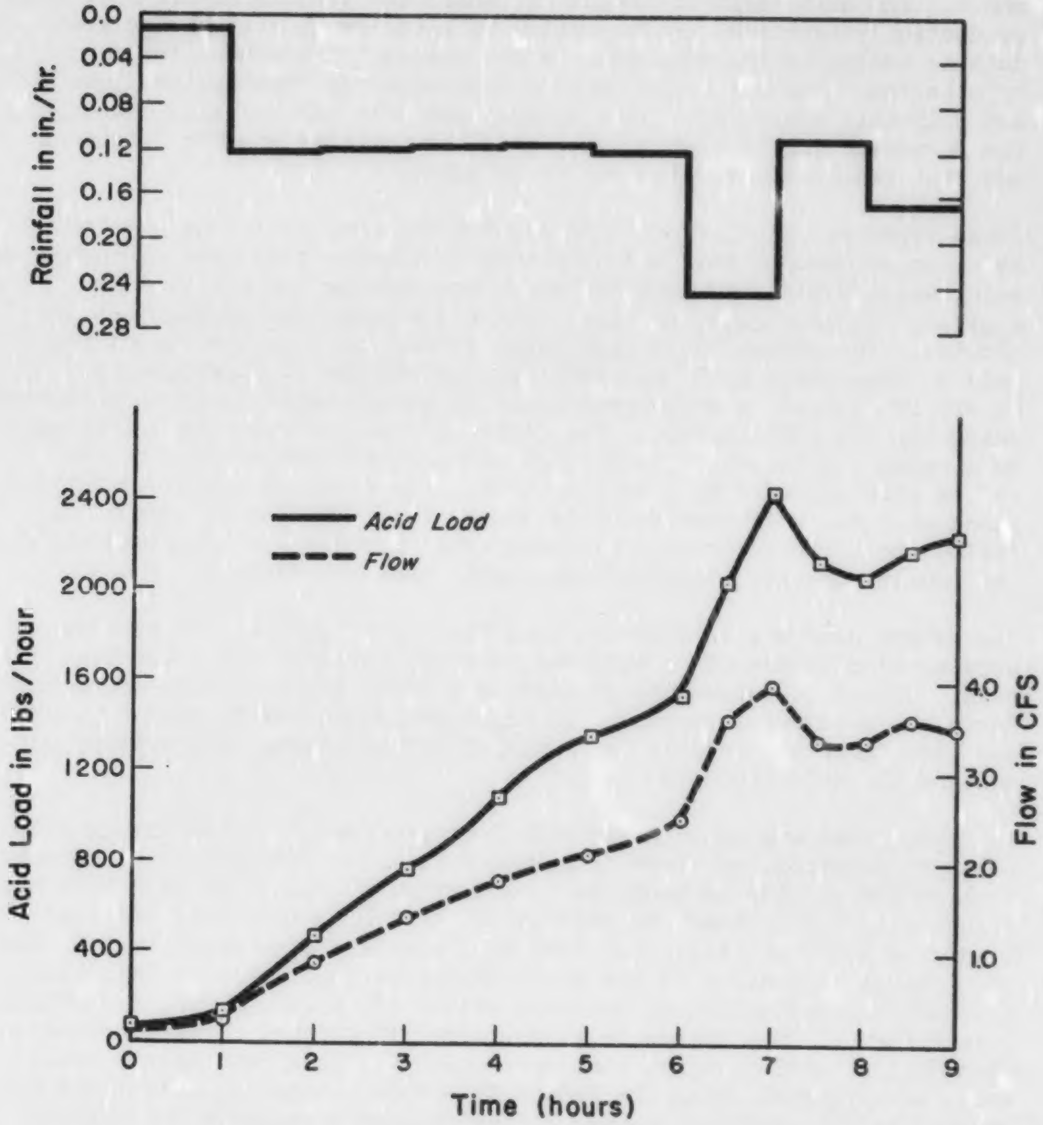


Figure B.3. Acid load for continuous rain

meaning; they are meant only to show the data format if the model were being applied to a specific refuse pile.

At the present time there is not sufficient data available to test the model. The data lacks in length of record and in determination of acid production parameters. A discussion will follow shortly giving what data is available and what data is yet needed. Therefore, for now, hypothetical data had to be applied (see topic on "Synthesized Data" later in this appendix). This synthesized data has no valid meaning, the purpose being to test and see that the computer program functions and that reasonable results can be obtained.

Three types of rain period, long drought followed by a long continuous rain, short drought followed by a long continuous rain, and a nonuniform rain, which would show that the model was working correctly, were applied. In each case, the acid load and stream flow hydrographs were plotted. The refuse model will print in tabular form the associated points (see Table B.13, Specific Day Output) for the hydrographs. Later, if desired, a subprogram could be incorporated to plot the curves using the IBM 1620 plotter. The first case was to find the acid load by 15-minute intervals if a long continuous rain was applied to the refuse pile preceded by a long drought. The selected storm was plotted along with the acid load curve in Figure B.3. The result appears to follow the expected trend. The load would increase to a maximum value and remain constant, provided the mantle were not depleted of acid.

The second case was to find the acid load if a long continuous rain were applied to the pile, with only a short interval since the last rain. Figure B.4 shows the results of a short interval continuous rain. Again the acid curve responded as might be predicted, in that it peaks and then falls off rapidly, a result of the preceding rain washing out most of the acid products.

The third test was to find the acid load produced by a nonuniform rain of short duration, as shown in Figure B.5. The acid load curve appears to give the results as would be anticipated. Thus, it is felt that the refuse pile model should be capable of reliably determining the acid load from a refuse pile. The lack of field data with which to test the model makes validation of the model structure impossible at this time. Although the assumption of constant values of ACDPRO, DEPTH, and SOLACD (see Table B.2) may appear to be oversimplifications, it is recommended that initial calibrations, when more complete field data become available, be attempted using the Refuse Pile Model variable definitions as defined in the preceding pages. If sufficiently accurate calibration is not possible, then the modified version of the Refuse Pile Model, referred to as the Combined Refuse Pile-Strip Mine Model (CRPSMM), is recommended. The CRPSMM modification of the basic Refuse Pile Model is described in a later portion of this appendix.

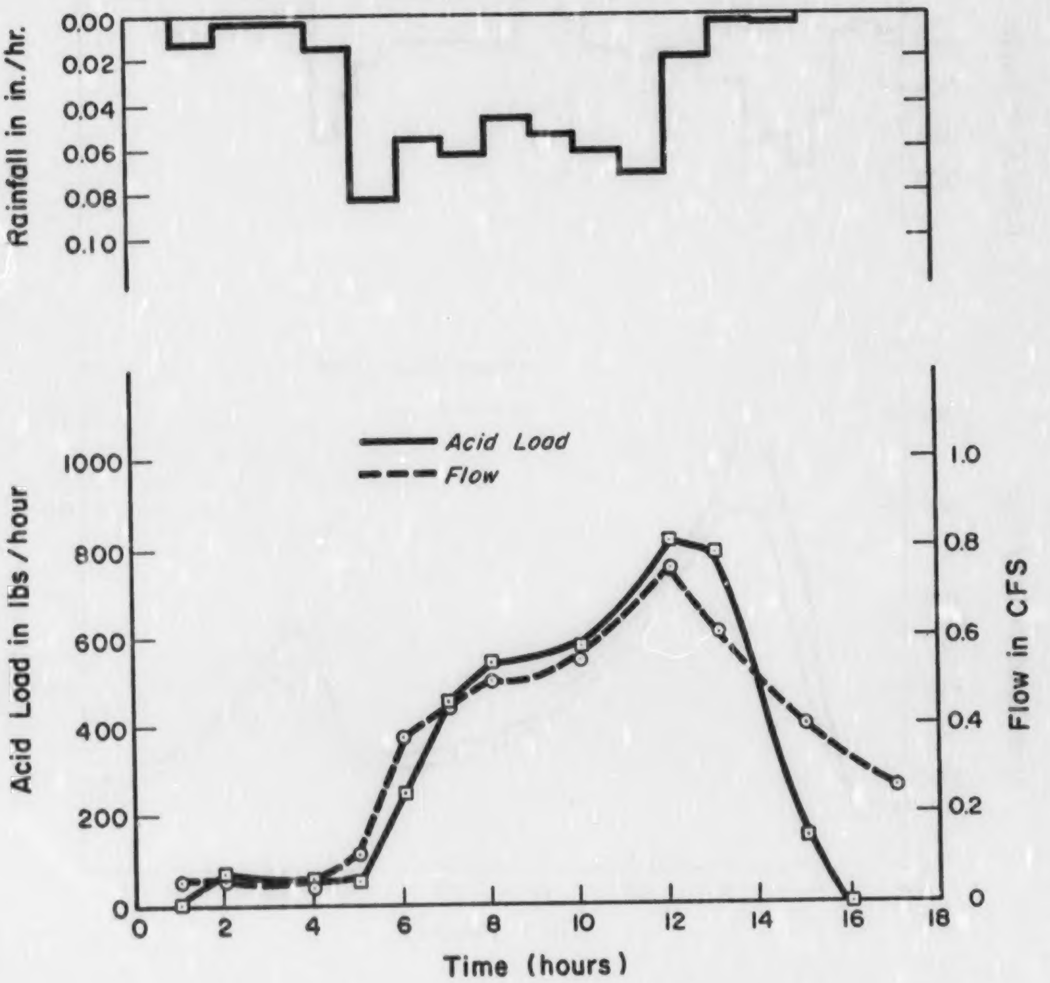


Figure B.4. Acid load for short interval continuous rain

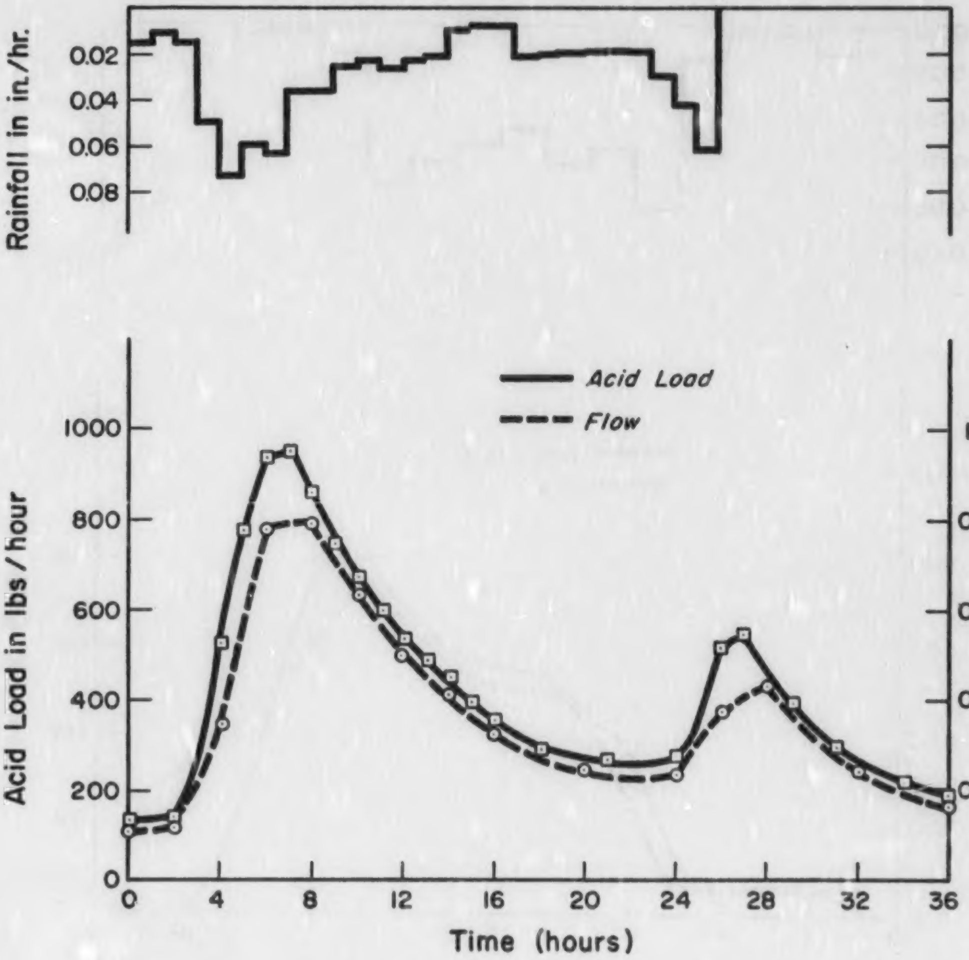


Figure B.5. Acid load for nonuniform short duration rain

The factors in the Refuse Pile Model should be set equal to unity and when sufficient field data is available the factors can be adjusted accordingly. With each experimental pile, the factors can then be set into general classes so that, eventually, the model will be able to predict the acid load based only on physical observations. Until that time, general guidelines will be given for each factor. The factors are listed below.

A detailed discussion of each of the factors will follow:

FACDEP - Factor for depth of outer mantle
FACDIR - Factor for amount of acid going into direct runoff
FACFB - Factor for amount of water coming from baseflow
FACFD - Factor for amount of water coming from direct runoff
FACFI - Factor for amount of water coming from interflow
FACINT - Factor for amount of acid going into interflow storage
FACLZ - Factor for amount of acid going into lower zone
FACRDS - Factor for amount of acid going into deep storage
FACREB - Factor for amount of acid going into channel by baseflow
FACRED - Factor for amount of acid going into channel by direct runoff
FACREI - Factor for amount of acid going into channel by interflow
FACUZ - Factor for amount of acid going into upper zone

The factor FACDEP is used to vary the amount of acid being produced in the outer mantle. The acid production rate was from a surface experiment and thus expresses a rate based on the unit depth. It is known that the production rate changes with depth, but in this model the rate was assumed constant through the outer mantle. Therefore, depending on the depth of the outer mantle, FACDEP should be increased to reflect the fact that more or less acid is being produced. If enough field data is available, this factor could be determined by looking at a long continuous rain that would wash all the products out of the pile. If the synthesized model runs out of acid products before the field data then the factor FACDEP should be increased.

The four factors FACDIR, FACINT, FACLZ, FACUZ all affect the amount of acid being taken by the precipitation into the four areas of direct runoff, interflow storage, upper zone and lower zone. If the pile is the average condition of the watershed, then the four factors might be equal to unity, but if this is not the case, the factors must be greater or less than unity. These factors involved the amount of acid transported to the four areas of the refuse pile. If, by field inspection, it is felt more water is going into one of the zones than the average condition of the watershed, then that factor should be increased. If the opposite is true, then the factor should be decreased.

Until ample true field data is available it would be best to set the factors equal to unity. Once actual detailed data is available the factors can be adjusted to reflect the true case.

To adjust the amount of acid leaving the pile by the four routes of direct runoff, interflow, baseflow and deep storage flow, four adjustment factors have been included FACRED, FACREL, FACREB and FACROS. These factors are dependent on the location of the refuse pile with respect to the receiving stream. Once adequate field data is available these factors can be adjusted to reflect the variable situation. If, by looking at the simulated acid hydrograph and the actual acid hydrograph obtained from a field refuse pile, it is noted that the simulated acid hydrograph's peak is not high enough, this would indicate more acid needs to be coming off in direct runoff, so FACRED would have to increase. FACREB would have to be increased if the simulated curve has less baseflow than the actual refuse pile. If as time increases it is noted that the simulated curve lags the true curve, FACREI should be increased. As an alternate, if the curve still is lagging, the factor for deep storage flow would have to decrease. If the reverse case is found for any of the preceding conditions, the factors should be adjusted accordingly.

The actual flow from the pile has three factors for adjustment. The factors FACFD, FACFI, and FALFB affect the flow coming from direct runoff, interflow, and baseflow. Again the actual and simulated hydrographs can be constructed and compared. The adjustments are the same as was outlined for the factors affecting the acid leaving the refuse pile.

Once these factors are set for a pile, the pile can simulate any condition desired.

A brief summary of the information required to use the model follows:

1. The streamflow simulation of the watershed from the Stanford Watershed Model.
2. The acid production rate.
3. Depth of the outer mantle.
4. The area extent of refuse pile.
5. The solubility of acid products.

With this data the Refuse Pile Model can simulate the acid load from a refuse pile. There are 12 adjustment factors in the model so that when sufficient field data is available the simulation can be adjusted until acceptable synthesis is obtained.

DATA FROM THE STANFORD WATERSHED MODEL

At the present time the Stanford Watershed Model has been tested using data obtained from the North Appalachian Experimental Watershed located near Coshocton, Ohio. This data is commonly known as the Watershed 94 data. Since this data was available it was chosen to be used as the input hydrologic data into the Refuse Pile Data. Because of the length of data (5 years) the input data will be kept on a 9 track magnetic tape. The tape label is APRPM1 and the slot number is M111.

The following will tell exactly what program statements in the Stanford Watershed Model must be added or deleted. Three new JCL card must be added after JCL statement number SM0010. (Reference 10 Part II) JCL cards are:

```
//Go.F To 3F001 DD DSN = STANDATA, UNIT = T360, LABEL = (1,SL),
          DISP = (NEW, KEEP),

//          VOL = (PRIVATE, RETAIN, SER = APRPM1),

//          DCB = (RECFM = FC, LRECL = 84, BLKSIZE = 3360)
```

In subroutine DYLOOP the following changes must be made:

Statement		Change To
0983		If(J.EQ.I. and .0023.EQ. 1) go to 55555
0984		Delete
0985		Delete
0986		Delete
0987		Delete
0988		55555 DAY = MOD(I, DDLM)
1000	Write	(3,129) DDYR1, FA, DAY, J, DD23, PR, ENTRUZ, ENTRLZ, RGX, OVFLST, DIRRNF, INTF, BASFLW, TOTFLW
1001		Delete
LV0120		129 FORMAT (I4, 4I2, 9F8.6)

Later two additions should be made:

1. Baseflow should be made equal to GWF.
2. ZTEMP should be added to statement 1000, for temperature option.

Synthesized Data

The following data was inputed into the refuse pile model in order to recreate the synthesized data for the three test cases of long drought followed by a long continuous rain, short drought followed by a long continuous rain, and a nonuniform rain.

In all three cases the hydrologic data was obtained from the Standard Watershed Model, applied to Watershed 94, near Coshocton, Ohio. The following data was constant in each case:

ACDPRO = 210.
SOLACO = 5000.
DEPTH = 11.
AREA = 6609666.
OPTI = 0.
FACFI = 1.

FACDEP = 1.
FACLZ = 1.
FACRED = 1.
FACREE = 1.
FACREB = 1.
FACFD = 1.
FACFB = 1.

Case 1: Acid Load Continuous Rain used the following input data

FACDIR = 1.
FACINT = 1.
FACUZ = 1.
FACRDS = 0.

YEAR 1 = 58
DAY 1 = 10
MONTH 1 = 4
NDAY = 30

The data plotted was from the twentieth day and twentieth hour to the twenty-first day and sixth hour.

Case 2: Acid Load Short Interval Continuous Rain used the following input data.

FACDIR = 0.01
FACINT = 0.1
FACUZ = 0.01
FACRDS = 1.

YEAR 1 = 58
DAY 1 = 10
MONTH 1 = 4
NDAY = 30

The data plotted was from the fifteenth day and sixth hour to the sixteenth day and first hour.

Case 3: Acid Load Nonuniform Short Duration Rain used the following data.

FACDIR = 1.
FACINT = 1.
FACUZ = 1.
FACROS = 0.

YEAR 1 = 59
DAY 1 = 11
MONTH 1 = 3
NDAY = 30

The data plotted was from the twelfth day and seventh hour to the thirteenth day and nineteenth hour.

REFUSE PILE COMPUTER PROGRAM

```
C  THIS IS THE COMPUTER PROGRAM FOR THE SIMULATION OF ACID MINE DRAINAGE
C       FROM A REFUSE PILE.  THE HYDROLOGY OF THE SYSTEM IS OBTAINED
C       FROM THE OHIO STATE UNIVERSITY VERSION OF THE STANFORD STREAM-
C       FLOW SIMULATION MODEL.  THIS MODEL WAS CONSTRUCTED AS PART OF
C       A MASTER THESIS.  THIS IS THE FEBRUARY     1973 VERSION.
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      REAL INTF
      INTEGER YEAR1,MONTH1,DAY1,YEAR2,MONTH2,DAY2,YEARED,DDYR1,FA,DAY,J,
     1DD23,MONED,YEARPR,YEARST,COUNT,DDAY,DAYOUT,DAYEND
      DIMENSION ARD(12,31),ARI(12,31),ARR(12,31),SAD(12),SAI(12),SAR(12)
     1,TACD(12,31),STACD(12),TSTR(12,31),STSTR(12),SDRR(12,31),SSDRR(12)
     2,SINT(12,31),SSINT(12),SMAS(12,31),SSMAS(12),DDAY(32)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  INITIALING THE AMOUNT OF ACID IN THE FOUR ZONES PLUS THE  TOTAL
C       AMOUNT OF ACID IN THE PILE
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      EXADIR=0.
      EXAINT=0.
      EXALZ=0.
      EXAUZ=0.
      AMTACD=0.
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  READING IN THE REFUSE PILE DATA:  1) THE ACID PRODUCTION RATE =
C       ACDPRO IN POUNDS OF ACIDITY PER ACRE-DAY  2) SOLUBILITY OF ACID
C       PRODUCTS = SOLACD IN MG PER LITER  3) DEPTH OF OUTER MANTLE =
C       DEPTH IN FEET  4) AREA OF REFUSE PILE = AREA IN SQUARE FEET
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,2001) ACDPRO,SOLACD,DEPTH,AREA
 2001 FORMAT(4F10.2)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  READING IN THE 12 ADJUSTMENT FACTORS: WILL BE CHANGED AS NEEDED FOR A
C       SPECIFIC REFUSE PILE
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,2002) FACDEP,FACDIR,FACINT,FACLZ,FACUZ,FACRED,FACREI,FACRER
     1,FACRDS,FACED,FACEI,FACER
 2002 FORMAT(12F5.4)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  IF YOU WANT TO RUN PROGRAM TO ADJUST THE ACID PRODUCTION RATE DUE TO
C       TEMPERATURE CHANGES THEN CALL FOR OPT1=1.  IF YOU CALL THIS
C       OPTION MAKE SURE THAT YOU READ IN THE TEMPERATURE VALUES WITH
C       THE OTHER INPUT SATA FROM THE STANFORD WATERSHED MODEL.  IF YOU
C       DO NOT WISH TO USE THIS OPTION THEN LET OPT1=0.
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      READ(5,2200) OPT1
 2200 FORMAT(F10.0)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHANGING SOLACD TO  POUNDS PER CUBIC FEET
C  FACTOR TO CHANGE INCHES OF RUNOFF OF REFUSE PILE TO CUBIC FEET PER
C       SECOND OF RUNOFF
C  FACTOR TO CHANGE INCHES OF RUNOFF FROM REFUSE PILE TO CUBIC FEET OF
C       RUNOFF (BASED ON 15 MINUTE INTERVAL)
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SOLACD= SOLACD*2.205/(35310)
      CFS=AREA*0.24*0.268888*0.0003587
      CF=CFS*60.*15.
C
```

Reproduced from
best available copy.

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C THE NEXT LEAP YEAR WILL OCCUR IN WHAT WATER YEAR - LAST TWO DIGITS
C       DIVIDED BY 4
C THE STARTING WATER YEAR OF THE INPUT DATA FROM THE STANFORD WATERSHED
C       MODEL
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      YEARLP=14.75
      YEARST=58
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C READING IF DATA FOR SPECIFIC DAY YOU WANT OUTPUT - SET FOR ONE TIME
C       PER WATER YEAR.  YEAR1= WATER YEAR, MONTH1 = MONTH, DAY1 = DAY
C       TO START OUTPUT, NDAY = NUMBER OF CONSECTIVE DAYS OF OUTPUT
C       REQUESTED
C       THE TIME IS BASED ON A WATER YEAR WHERE THE FIRST MONTH IS
C       OCTOBER
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 9100 READ(5,2000) YEAR1,MONTH1,DAY1,NDAY
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C CHECKING IF ENOUGH DATA HAS BEEN OUTPUTED AS REQUESTED:  IF SO
C       WILL STOP
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(YEAR1 .EQ. 0) GO TO 9200
 2000 FORMAT(4I5)
      DAYEND=NDAY*96
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C FOR EACH WATER YEAR INITILIZE THE DAILY, MONTHLY, AND YEARLY VALUES.
C       SET EVERY THING EQUAL TO ZERO
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DAYOUT=1
 8900 DDAY(1)=1.
      DO 200 I=1,12
      DO 201 J=1,31
      ARD(I,J)=0.
      ARI(I,J)=0.
      DDAY(J+1)=DDAY(J)+1.
      TACO(I,J)=0.
      TSTR(I,J)=0.
      SDRR(I,J)=0.
      SINT(I,J)=0.
      SRAS(I,J)=0.
  201 ARB(I,J)=0.
      SAD(I)=0.
      SAI(I)=0.
      STACO(I)=0.
      STSTR(I)=0.
      SSDRR(I)=0.
      SSINT(I)=0.
      SSRAS(I)=0.
  200 SAB(I)=0.
      SUMAD=0.
      SUMAI=0.
      SUMAR=0.
      SUMTA=0.
      SUMST=0.
      SUMSR=0.
      SUMSI=0.
      SUMSD=0.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C  PROCEDURE SO THAT THE EXTRA DAYS IN THE MONTH WILL CREATE AN OVER-
C        FLOW IN THE OUTPUT DATA.  WILL ADJUST FOR THE LEAP YEAR.
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SLYEAR=FLOAT(YEARST)/4.
      COUNT=0.
      J=2
 7000 I=31
 7001 ARD(J,I)=100000000.
      ARR(J,I)=100000000.
      ARI(J,I)=100000000.
      TACD(J,I)=100000000.
      TSTR(J,I)=100000000.
      SDRR(J,I)=100000000.
      SINT(J,I)=100000000.
      SBAS(J,I)=100000000.
      COUNT=COUNT+1.
      IF(COUNT .EQ. 1.) J=7
      IF(COUNT .EQ. 3.) J=12
      IF(COUNT .EQ. 2.) J=9
      IF(COUNT .EQ. 4.) J=5
      IF(COUNT .EQ. 5.) GO TO 7004
      IF(COUNT .EQ. 6. .AND. SLYEAR .EQ. YEARLP) GO TO 7002
      IF (COUNT .EQ. 6. ) GO TO 7003
      IF(COUNT .EQ. 7.) GO TO 7005
      GO TO 7000
 7004 I=30
      GO TO 7001
 7002 YEARLP=YEARLP+1.
      GO TO 7005
 7003 I=29
      GO TO 7001
 7005 CONTINUE
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  READING IN HYDROLOGIC DATA FROM THE STANFORD WATERSHED MODEL.  THIS
C        INCLUDES THE RAINFALL DEPOSITION AND THE STREAMFLOW.  THE TIME
C        IS BASE ON A WATER YEAR  IF  OCTOBER FIRST TO SEPTEMBER  30TH.
C        THE VARAIBLES ARE THE YEAR = DDYR1, MONTH = FA, DAY = DAY,
C        HOUR = J, HOUR INTERVAL = DD23, THE RAIN = PR IN INCHES, THE
C        WATER GOING TO UPPER ZONE = ENTRUZ IN INCHES, THE WATER GOING
C        TO LOWER ZONE = ENTRLZ IN INCHES, THE STREAMFLOW COMING FROM
C        DIRECT RUNOFF = DIRRNF IN INCHES, THE STREAMFLOW COMING FROM
C        INTERFLOW = INTF IN INCHES, THE STREAMFLOW COMING FROM BASEFLOW
C        = BASFLW IN INCHES, AND THE TOTAL STREAMFLOW = TOTFLW IN INCHES.
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 8800 READ(3,129)  DDYR1,FA,DAY,J,DD23,PR,ENTRUZ,ENTRLZ,RGX,OVFLST,
     1 DIRRNF,INTF,BASFLW,TOTFLW
  129 FORMAT(I4,4I2,9F8.6)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  ADJUSTING ACID PRODUCTION RATE DUE TO TEMPERATURE CHANGE
C  TEMPERATURE SHOULD BE IN DEGREES FAHRENHEIT
C  THE ADJUSTMENT IS THE RULE OF THUMB -- THE RATE DOUBLES FOR A 10
C        DEGREE CENTIGRADE INCREASE
C   TO IS THE TEMPERATURE AT WHICH THE ACID PRODUCTION RATE WAS
C        DETERMINED.
C  T IS THE INPUT TEMPERTURE FROM THE STANFORD MODEL
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      TO=77.
      IF(OPT1 .EQ. 0.) GO TO 5555
      CK=2.**((T-TO)/18.)
      GO TO 5556
```

```
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  IF CK=1. THEN THE ACID PRODUCTION IS NOT BEING ADJUSTED FOR
C        TEMPERATURE CHANGES
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 5555 CK=1.
 5556 ACDPRO=ACDPRO*CK
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  TIME IS 15 MINUTES OR 0.25 HOUR.  MUST CORRESPOND WITH THE TIME
C        INTERVAL FROM THE STANFORD WATERSHED MODEL
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      TIME=0.25
      IF(PR .NE. 0.0) TIME=0.
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CALCULATING THE AMOUNT OF ACID BEING PRODUCED
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      AMTACD=ACDPRO*TIME*AREA*DEPTH*FACDEP/(24.      *43500.)+AMTACD
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CALCULATING THE AMOUNT OF ACID BEING REMOVE DURING THE PRECIPITATION
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ACDDIR=FACDIR*SOLACD*OVFLST*CF
      ACDINT=FACINT*SOLACD*RGX*CF
      ACDLZ=FACLZ*SOLACD*ENTRLZ*CF
      ACDUZ=FACUZ*SOLACD*ENTRUZ*CF
      TOTAL=AMTACD-ACDINT-ACDLZ-ACDUZ-ACDDIR
      IF(TOTAL .LE. 0.0)GO TO 60
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  STORING ACID IN FOUR ZONES
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      EXADIR=ACDDIR+EXADIR
      EXAINT=ACDINT+EXAINT
      EXALZ=ACDLZ+EXALZ
      EXAUZ=ACDUZ+EXAUZ
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  REMOVING THE ACID
C  CHECKING TO MAKE SURE THERE IS ACID TO BE REMOVED
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   60 IF(DIRRNF .EQ. 0.0) GO TO 22
      AREDIR=DIRRNF*SOLACD*FACRED*CF
      IF(EXADIR .LE. 0.)GO TO 21
      IF(EXADIR .LE. AREDIR) AREDIR=EXADIR
      EXADIR=EXADIR-AREDIR
      GO TO 23
   21 IF(EXAUZ .LE. 0.) GO TO 22
      IF(EXAUZ .LE. AREDIR) AREDIR=EXAUZ
      EXAUZ=EXAUZ-AREDIR
      GO TO 23
   22 AREDIR=0.0
   23 IF(INTF .EQ. 0.0) GO TO 32
      AREINF=INTF*SOLACD*FACRFI*CF
      IF(EXAINT .LE. 0.) GO TO 31
      IF(EXAINT .LE. AREINF) AREINF=EXAINT
      EXAINT=EXAINT-AREINF
      GO TO 30
   31 IF(EXALZ .LE. 0.) GO TO 32
      IF(EXALZ .LE. AREINF) AREINF=EXALZ
      EXALZ=EXALZ-AREINF
      GO TO 30
   32 AREINF=0.
   30 IF(BASFLW .EQ. 0.0) GO TO 43
      AREBFL=BASFLW*SOLACD*FACREB*CF
      IF(EXALZ .LE. 0.) GO TO 43
      IF(EXALZ .LE. AREBFL) AREBFL=EXALZ
      EXALZ=EXALZ-AREBFL
      GO TO 40
   43 AREBFL=0.
   40 IF(INTF .EQ. 0.0) GO TO 51
      AREDSR=INTF*SOLACD*FACRDS*CF
      IF(EXALZ .LE. 0.) GO TO 52
      IF(EXALZ .LE. AREDSR) AREDSR=EXALZ
      EXALZ=EXALZ-AREDSR
      GO TO 51
   52 AREDSR=0.
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  FINDING THE DAILY VALUES
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   51 CFDIR=DIRRUF*CFS*FACFD
      CFINT=INTF*CFS*FACFI
      CFBAS=BASFLW*CFS*FACFB
      CFTOT=CFBAS+CFINT+CFDIR
      ARETOT=AREDIR+AREINF+AREBFL
      ARD(FA,DAY)=AREDIR+ARD(FA,DAY)
      ARI(FA,DAY)=AREINF+ARI(FA,DAY)
      ARB(FA,DAY)=AREBFL+ARB(FA,DAY)
      SBAS(FA,DAY)=CFBAS+SBAS(FA,DAY)
      SINT(FA,DAY)=CFINT+SINT(FA,DAY)
      SDRR(FA,DAY)=CFDIR+SDRR(FA,DAY)
      TSTR(FA,DAY)=CFTOT+TSTR(FA,DAY)
      TACD(FA,DAY)=AREDIR+AREINF+AREBFL+TACD(FA,DAY)
   65 IF(YEAR1 .EQ. DDYR1) GO TO 120
      GO TO 100
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHECKING IF THIS IS SPECIFIC DAY TO BE OUTPUTED
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  120 IF(MONTH1 .EQ. FA .OR. DAYOUT .GT. 1) GO TO 130
      GO TO 100
  130 IF(DAY1 .EQ. DAY .OR. DAYOUT .GT. 1) GO TO 131
      GO TO 100
  131 IF(DAYOUT .EQ. 1) WRITE(6,3030)
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHECKING IF THIS IS THE END OF THE SPECIFIC DAY OUTPUT
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(DAYOUT .GT. DAYEND) GO TO 100
      DAYOUT=DAYOUT+1
 3030 FORMAT('1',94H YR MO DY HR PD    AREDIR     AREINF     AREBFL     ARE
     1TOT     CFDIR     CFINT     CFBAS     CFTOT)
      WRITE(6,3031) DDYR1,FA,DAY,J,DD23,AREDIR,AREINF,AREBFL,ARETOT,CFD
     1IR,CFINT,CFBAS,CFTOT
 3031 FORMAT(' ',5(1X,I2),8(1X,F9.4))
  100 YEARPR=YEARST+1
      IF(YEARPR .EQ. DDYR1 .AND. FA .EQ. 1) GO TO 8000
      GO TO 8800
C
C  * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  FINDING THE MONTHLY VALUES
C  FINDING YEARLY VALUES
```

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 8000 DO 700 I=1,12
      DO 701 J=1,31
      IF(ARD(I,J) .EQ. 100000000.) GO TO 702
      SAD(I)=ARD(I,J)+SAD(I)
      SAI(I)=ARI(I,J)+SAI(I)
      SAB(I)=ARB(I,J)+SAB(I)
      SSBAS(I)=SBAS(I,J)+SSBAS(I)
      SSINT(I)=SINT(I,J)+SSINT(I)
      SSDRR(I)=SDRR(I,J)+SSDRR(I)
      STSTR(I)=TSTR(I,J)+STSTR(I)
      STACD(I)=TACD(I,J)+STACD(I)
  702 CONTINUE
  701 CONTINUE
      SUMAD=SAD(I)+SUMAD
      SUMAI=SAI(I)+SUMAI
      SUMAB=SAB(I)+SUMAB
      SUMTA=STACD(I)+SUMTA
      SUMST=STSTR(I)+SUMST
      SUMSB=SSBAS(I)+SUMSB
      SUMSI=SSINT(I)+SUMSI
      SUMSD=SSDRR(I)+SUMSD
  700 CONTINUE
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  OUTPUTING THE 'B' DAILY TABLES PLUS THE YEARLY SUMMARY
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3003)
 3003 FORMAT(' ',38X,48HSYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS
     1)
      WRITE(6,3013)
      DO 300 I=1,31
      WRITE(6,3011) DDAY(I),(ARD(J,I),J=1,12)
  300 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3002)
 3002 FORMAT(' ',39X,44HSYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS)
      WRITE(6,3013)
      DO 301 I=1,31
      WRITE(6,3011) DDAY(I),(ARI(J,I),J=1,12)
  301 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3001)
 3001 FORMAT(' ',39X,43HSYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS)
      WRITE(6,3013)
      DO 302 I=1,31
      WRITE(6,3011) DDAY(I),(ARB(J,I),J=1,12)
  302 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3004)
 3004 FORMAT(' ',42X,37HSYNTHESIZED TOTAL ACID LOAD IN POUNDS)
      WRITE(6,3013)
      DO 303 I=1,31
      WRITE(6,3011) DDAY(I),(TACD(J,I),J=1,12)
  303 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3006)
 3006 FORMAT(' ',35X,50HSYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECO
     1ND)
      WRITE(6,3013)
      DO 305 I=1,31
      WRITE(6,3020) DDAY(I),(SDRR(J,I),J=1,12)
  305 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3007)
 3007 FORMAT(' ',37X,46HSYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 306 I=1,31
      WRITE(6,3020) DDAY(I),(SINT(J,I),J=1,12)
  306 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3008)
 3008 FORMAT(' ',37X,45HSYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 307 I=1,31
      WRITE(6,3020) DDAY(I),(SBAS(J,I),J=1,12)
  307 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3005)
 3005 FORMAT(' ',37X,47HSYNTHESIZED STREAMFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 304 I=1,31
      WRITE(6,3020) DDAY(I),(TSTR(J,I),J=1,12)
  304 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,4005)
 4005 FORMAT('-',3X,3HOCT,6X,3HNOV,6X,3HDEC,6X,3HJAN,6X,3HFEB,6X,3HMAR,6
     1X,3HAPR,6X,3HMAY,6X,3HJUN,6X,3HJUL,6X,3HAUG,6X,3HSEP,3X,10HYEAR TO
     2TAL)
      WRITE(6,4000)
      WRITE(6,4000)
      WRITE(6,4001)
 4001 FORMAT('0',48HSYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS)
      WRITE(6,4002) (SAD(I),I=1,12),SUMAD
      WRITE(6,4004)
 4004 FORMAT('0',44HSYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS)
      WRITE(6,4002) (SAI(I),I=1,12),SUMAI
      WRITE(6,4006)
 4006 FORMAT('0',43HSYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS)
      WRITE(6,4002) (SAB(I),I=1,12),SUMAB
      WRITE(6,4007)
 4007 FORMAT('0',37HSYNTHESIZED TOTAL ACID LOAD IN POUNDS)
      WRITE(6,4002) (STACD(I),I=1,12),SUMTA
      WRITE(6,4008)
 4008 FORMAT('0',50HSYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (SSDRR(I),I=1,12),SUMSD
      WRITE(6,4010)
 4010 FORMAT('0',46HSYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (SSINT(I),I=1,12),SUMSI
      WRITE(6,4011)
 4011 FORMAT('0',45HSYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (SSBAS(I),I=1,12),SUMSB
      WRITE(6,4012)
 4012 FORMAT('0',78HSYNTHESIZED TOTAL AMOUNT OF WATER ENTERING THE STREA
     1M IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (STSTR(I),I=1,12),SUMST
 3011 FORMAT(' ',3X,I2,12(1X,F9.1))
 3012 FORMAT('1',40X,32HANNUAL SUMMARY FOR WATER YEAR 19,I2,1X,1H-,1X,
     12H19,I2)
 3013 FORMAT('-',3X,3HDAY,5X,3HOCT,6X,3HNOV,6X,3HDEC,6X,3HJAN,6X,3HFEB,6
     1X,3HMAR,6X,3HAPR,6X,3HMAY,6X,3HJUN,6X,3HJUL,6X,3HAUG,6X,3HSEP)
 3020 FORMAT(' ',3X,I2,12(1X,F8.4))
 4000 FORMAT(' ')
 4002 FORMAT(' ',12(1X,F8.0),1X,F11.0)
 4009 FORMAT(' ',12(1X,F8.0),1X,F11.0)
      YEARST=YEARST+1
 9000 GO TO 9100
 9200 STOP
      END
```

DESCRIPTION OF THE COMBINED REFUSE PILE - STRIP MINE MODEL (CRPSMM)

The Refuse Pile Model described in the preceding sections is intended for use in cases where there is a high rate of acid production at the surface of a refuse pile, or strip mine spoil bank. In cases where acid production rates are relatively low, or where acid producing strata may lie buried beneath a layer of relatively inert material, then the simplifying assumptions of the Refuse Pile Model relating to acid production and acid removal rates may not be sufficient to allow calibration of the model within the limits of accuracy required. For such cases, the Combined Refuse Pile - Strip Mine Model has been developed. The CRPSMM differs from the Refuse Pile Model primarily in the handling of acid production and acid removal simulation, with these refinements superimposed on the basic Refuse Pile Model. For a more detailed disucssion of the CRPSMM, the reader is referred to Maupin.[19]

ACDPRO Program Description

So that the Refuse Pile Model may be extended to include the simulation of strip mined areas and refuse piles, and to provide for pyrite oxidation rate-determining factors other than temperature, the subroutine ACDPRO has been developed to replace the constant acid production rate used in the Refuse Pile Model. A second major modification in the acid removal simulation will be described later.

From the standpoint of precise simulation, an ideal approach would be to use a three-dimensional finite difference unsteady state model of oxygen diffusion and reaction in the spoil or refuse. However, the computer program for such a model would require inordinately long computer run times even if adequate soil profile information were available to build and adjust such a model.

The alternative chosen for this study is a steady-state model of oxygen diffusion and reaction in a column of soil of unit surface area, with diffusion constrained to the vertical direction only. By measuring the surface areas of a representative acid producing region in a watershed, and multiplying the areas by the appropriate specific acid production rate (corresponding to similar soil columns), an adequate estimate for the entire composite area can be calculated. The assumption of steady-state is considered to provide a sufficiently accurate estimate of the average acid production rate since, in the simulation, acid production is stopped during periods of rainfall, and the error of stopping the reaction too soon (before rainfall infiltration lowers oxygen diffusivity and stops the reaction) is somewhat balanced by the error of restarting the reaction too soon (before the soil moisture has drained sufficiently to allow oxygen diffusion again). The assumption of oxygen diffusion in the vertical direction only is justified since the areas to be modeled are large (usually several acres in size) and can be considered as infinite plates on which the edge effects may be neglected.

ACDPRO is the computer subroutine developed to model the acid production of a generalized column of soil as shown in Figure B.6. The generalized column is divided into two sections, a pyrite layer overlain by an inert soil layer, which may or may not be present. The basic strategy of the solution is to find by iteration of steady state oxygen concentration at the interfacial boundary between the two layers (X_{A_2}) which, in turn, allows calculation of the rate of reaction in the reactive layer. This is possible since at steady-state the rate of oxygen diffusion through the inert layer is equal to the rate of pyrite oxidation in the pyrite layer. In the special case of no inert layer, the known atmospheric oxygen concentration at the air-pyrite interface again allows computation of the overall oxidation rate.

For the pyrite layer, the equation of continuity (Bird et al.[20]) in cylindrical coordinates is

$$\frac{\partial C_A}{\partial t} + \left(v_r \frac{\partial C_A}{\partial r} + v_\theta \frac{1}{r} \frac{\partial C_A}{\partial \theta} + v_z \frac{\partial C_A}{\partial z}\right) = D_{AB}\left(\frac{1}{r} \frac{\partial}{\partial r}\left(r \frac{\partial C_A}{\partial r}\right) + \frac{1}{r^2} \frac{\partial^2 C_A}{\partial \theta^2} + \frac{\partial^2 C_A}{\partial z^2}\right) + R_A \qquad \text{(B.22)}$$

where C_A = the concentration of oxygen (m/ℓ^3),
r = the radial dimension (ℓ),
θ = the angular dimension (ℓ),
z = the vertical dimension (ℓ),
v = the bulk gas velocity (ℓ/t),
D_{AB} = the diffusivity of O_2 in the other gases present (ℓ^2/t), and
R = the volumetric generation term for oxygen $\left(\frac{m}{\ell^3 - t}\right)$.

By applying the assumptions of steady-state diffusion in the vertical direction only, all derivatives with respect to time (t) and the angular and radial direction (ℓ) are set equal to zero. Since diffusion is through a porous solid, an effective diffusivity (D_{eff}) is substituted for D_{AB} and the first order specific reaction rate equation ($Ae^{-\Delta E/RT}C_A$) is substituted for R_A. Equation (B.22) then becomes

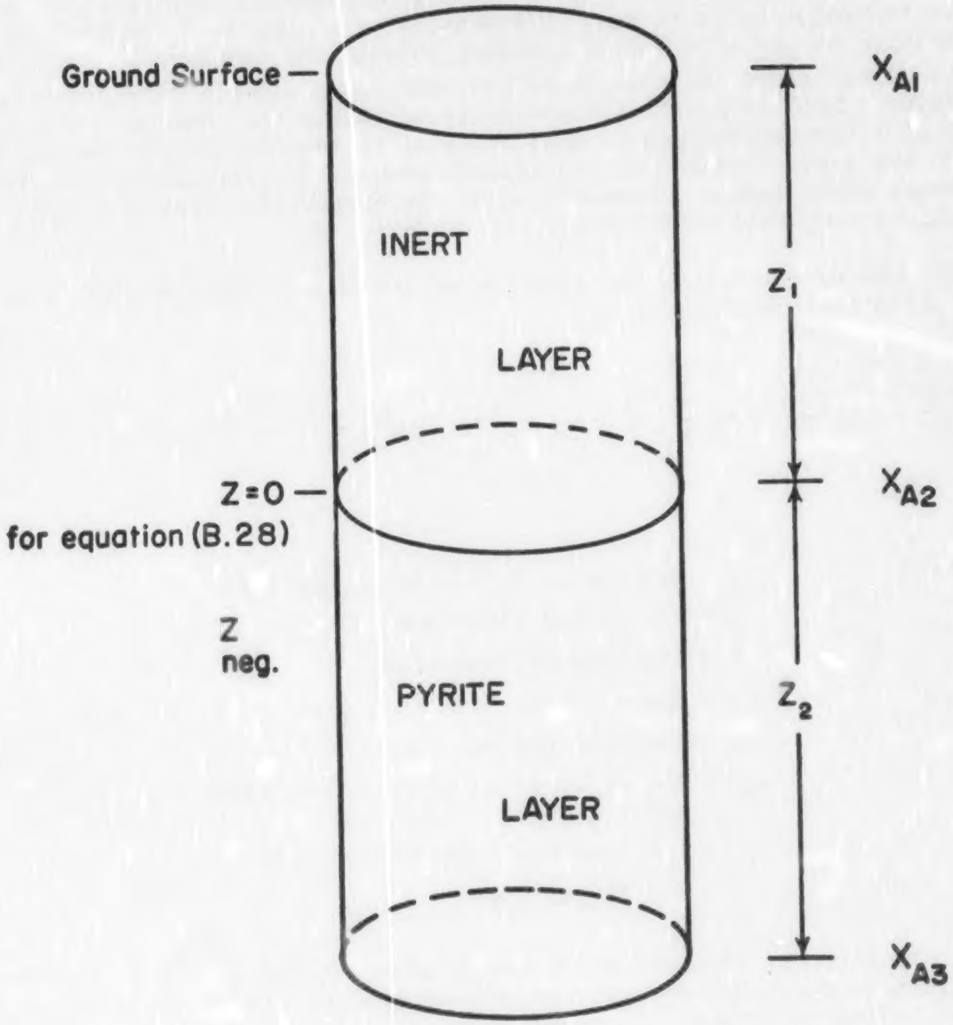

Ground Surface
X_{A1}
INERT
LAYER
Z_1
Z=0
for equation (B.28)
X_{A2}
Z
neg.
PYRITE
LAYER
Z_2
X_{A3}

Figure B.6. Soil column

$$v_z \frac{dC_A}{dz} = D_{eff} \frac{d^2C_A}{dz} + Ae^{-\frac{\Delta E}{RT}} C_A \qquad \text{(B.23)}$$

which is a second order differential equation with a variable coefficient v_z. The term $v_z \frac{dC_A}{dz}$ accounts for the bulk flow of oxygen through the reaction layer (this is also known as enhanced diffusion). A finite difference solution of equation (B.23) has shown that the enhanced diffusion term accounts for approximately 20% of the total reaction rate for the Column. This error is decreased if the upper surface of the pyrite layer is not exposed to the full atmospheric oxygen concentration. If the enhanced diffusion term is dropped, an analytical solution of equation (B.23) is made possible. From a practical standpoint, this is fully justifiable since the error thus introduced can be compensated by using a higher diffusivity or higher reaction rate constant and in any case the program variables must be further adjusted to simulate any available recorded data when the model is applied to a watershed. The simplified equation becomes

$$D_{eff} \frac{d^2C_A}{dz^2} + Ae^{-\frac{\Delta E}{RT}} C_A = 0 \qquad \text{(B.24)}$$

The analytical solution to this equation is given below along with an approximate solution (equation (B.25)) which will be used in this work. Equation (B.24) can be rewritten as

$$\frac{d^2C_A}{dz^2} - \alpha^2 C_A = 0$$

since the reaction term for oxygen consumption is negative and where α^2 is $Ae^{-\frac{\Delta E}{RT}}/D_{eff}$.

The general solution to equation (B.24) is

$$C_A = C_1e^{\alpha z} + C_2e^{-\alpha z}$$

If the constants of integration C_1 and C_2 are evaluated using the boundary conditions

$$C_A = C_{A2} \quad \text{at} \quad z = 0$$

$$\frac{dC_A}{dz} = 0 \quad \text{at} \quad z = z_L$$

the solution is

$$C_A = C_{A2}\left(\frac{1}{1 + e^{2z_L\alpha}}\right)e^{\alpha z} + C_{A2}\left(1 - \frac{1}{1 + e^{2z_L\alpha}}\right)e^{-\alpha z}$$

If instead the boundary conditions are approximated as

$$\frac{dC_A}{dx} = 0 \quad \text{at} \quad z = -\infty$$

the solution is

$$C_A = C_{A2}\, e^{\alpha z} \tag{B.25}$$

This approximation is much more readily handled by hand calculation or in an iterative computer solution, and has a maximum error of 17%.[19] This approximation is justified since the model must be adjusted to the acid production area based on field data.

The description of diffusion through the inert layer is based on the equation for diffusion through a stagnant gas film given by Bird et al.[20]

$$N_{AZ} = \frac{c\, D_{AB}(x_{A1} - x_{A2})}{(Z_2 - Z_1)(x_B)_{\ell m}} \tag{B.26}$$

where N_{AZ} = the rate of mass transfer through the gas film,

c = the total concentration (lb. moles/ft^3),

X_{A1} = the mole fraction oxygen in the atmosphere,

X_{A2} = the mole fraction oxygen at the interface,

$(Z_2 - Z_1)$ = gas film thickness (ft)

$(X_B)_{\ell m} = \dfrac{X_{B2} - X_{B1}}{\ell m \left(\dfrac{X_{B2}}{X_{B1}}\right)}$ = log mean mole fraction of all gaseous components other than oxygen

X_{B2} = the mole fraction of other gases at the interface, and

X_{B1} = the mole fraction of other gases at the surface.

It should be noted that this equation accounts for the enhanced diffusion of oxygen in the inert layer.

Again an effective diffusivity D_{eff} is substituted for D_{AB} and Z_1 is substituted for $(Z_2 - Z_1)$ in equation (B.26), giving

$$N_{AZ} = \frac{c\, D_{eff}(X_{A1} - X_{A2})}{Z_1 (X_B)} \qquad \text{(B.27)}$$

At steady-state N_{AZ} is equal to the rate of oxygen consumed in the pyrite layer for a soil column of unit area. The consumption rate in the pyrite layer can be calculated independently by integrating equation (B.25) over the thickness of the pyrite layer,

$$R_A \Delta Z = k \int_0^{Z_2} c\, X_{A2}\, e^{\sqrt{k/D_{eff}}\, Z} dz$$

$$R_A \Delta Z = k\, c\, X_{A2} \left[\frac{e^{\sqrt{k/D_{eff}}\, Z}}{\sqrt{k/D_{eff}}} \right] \qquad \text{(B.28)}$$

thereby allowing for the solution of X_{A2} by iteration. An initial estimated value of X_{A2} is assumed and $R_A Z$ is calculated using equation (B.28). $R_A \Delta Z$ is set equal to the flux N_{AZ} and a new value X_{A2} is calculated by equation (B.27). If the estimated and calculated values of

X_{A2} are not equal, an improved estimate is calculated based on the Wegstein convergence method as given by

$$X(n+1) = \frac{X(n-1)*Y(n) - Y(n-1)*X(n)}{X(n-1) - X(n) + Y(n) - Y(n-1)} \tag{B.29}$$

where $X(n+1)$ = the improved estimate of X_{A2},

$X(n)$ = the current estimate of X_{A2},

$X(n-1)$ = the previous estimate of X_{A2},

$Y(n)$ = the calculated value of X_{A2} using estimate $X(n)$, and

$Y(n-1)$ = the calculated value of X_{A2} using estimate $X(n-1)$.

The computer listing of ACDPRO is given in Figure B.7, and a flow chart of the program is given in Figure B.8. It should be noted that if there is no inert cover the program solves directly for the oxidation rate using equation (B.28). After the interfacial mole fraction (X_{A2}) is calculated the amount of acidity produced is computed, together with the mole fraction of oxygen at the lower surface of the pyrite layer. DO, an input variable having the dimension of length, is then introduced to separate the acid produced into two layers, a top layer which can be leached by direct runoff, and a lower layer which is leached by water going to interflow or baseflow only. The calculation of the rate of acid production above DO is calculated by setting Z_2 equal to the depth of pyrite above depth DO and solving equation (B.28) using the calculated interfacial mole fraction (X_{A2}). The acid production rate below DO is then obtained by difference.

By using this simulation program, acid production can be made responsive to changes in the following variables:

(1) Depth of inert cover
(2) Thickness of pyrite layer
(3) Diffusivity of both pyrite layer and inert layer
(4) Soil temperature
(5) Total pressure
(6) Reactivity of pyrite

ACDPRO inputs and outputs are listed in Table B.14. The inputs R and DEOR are known. Z_1, Z_2, P, T, and X_{A1} can be measured in the field and adjusted if necessary. The variables DOZ, DOZA, A, and DO will probably

```
C     ACID PRODUCTION SUBROUTINE
C         Z1=ZERO FOR EXPOSED PYRITE
      SUBROUTINE ACDPRO
      COMMON/AMM2/Z1,Z2,P,R,T,DOZ,A,DO,XA1,DEOR,DOZA,XA2,XA3,AP,AP1,DIFF
     *,NR,APO
      DIMENSION XA(25),YA(25)
      Z1=-Z2
      C=P/(R*T)
      Y1=A/EXP(DEOR/T)
  910 XA2=.10
      XB1=1.0-XA1
      NR=1
      NC=0
      Z=SQRT(Y1/DOZA)
      DIFF=0.
      IF(Z1.LE..0001)GO TO 10
  920 XB2=1.0-XA2
  925 XBN=(XB2-XB1)/ALOG(XB2/XB1)
      RA=Y1*XA2*C/Z
      RA3=1.0-EXP(Z*Z1)
      RAZ=RA*RA3
      XA2N=(C*DOZ*XA1-(    Z1)*XBN*RAZ)/(C*DOZ)
      X=XA2
      Y=XA2N
      IF(ABS((X-Y)/(X+Y)).LT..1E-9)GO TO 6
      IF(NR.GT.20)GO TO 6
      IF(NC.LE.1) GO TO 5
      IF(ABS(XA(NR)-X+Y-YA(NR)).LT..1E-9)GO TO 6
      XT=(XA(NR)*Y-YA(NR)*X)/(XA(NR)-X+Y-YA(NR))
      NR=NR+1
      XA(NR)=X
      YA(NR)=Y
      XA2=XT
      GO TO 920
   10 XA2N=XA1
      XA2=XA1
      RA=Y1*XA2*C/Z
      RA3=1.0-EXP(Z*Z1)
      RAZ=RA*RA3
      GO TO 6
    5 XA(NR)=X
      YA(NR)=Y
      XA2=Y
      NC=2
      GO TO 920
    6 XA2=XA2N
      AP=RAZ*55.7
      IF(DO-Z1)200,200,210
  200 APO=0.
      GO TO 220
  210 ZD=DO-Z1
      IF(ZD.GT.Z2)ZD=Z2
      ZD=-ZD
      RA4=1.-EXP(Z*ZD)
      APO=55.7*RA*RA4
  220 AP1=AP-APO
      IF(Z1.LE..0001)GO TO 20
      DAZ=(C*DOZ*(XA1-XA2)/(Z1*XBN))
      DIFF=RAZ-DAZ
   20 XA3=XA2*EXP(Z*Z1)
 1001 RETURN
      END
```

Figure B.7. ACDPRO Program listing

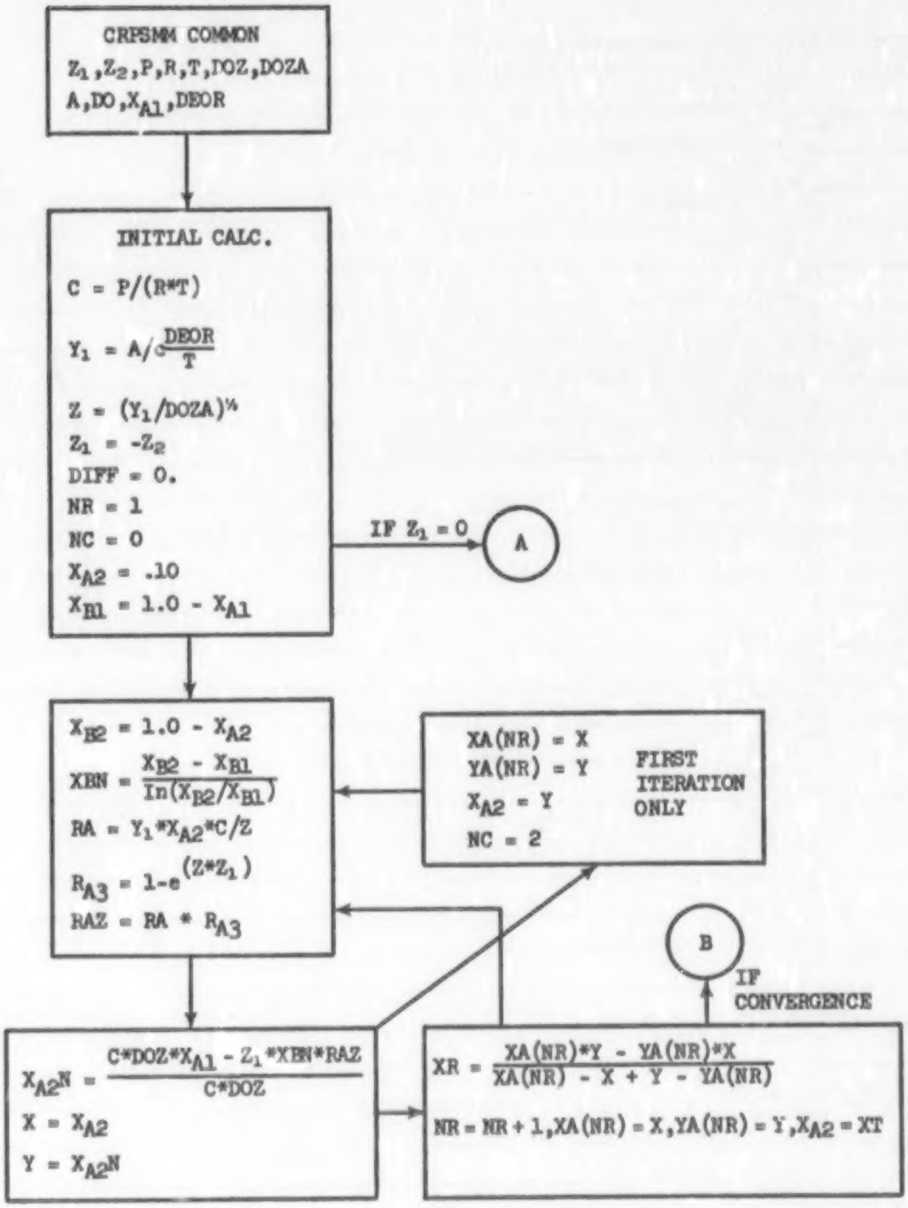

CRPSMM COMMON
Z1,Z2,P,R,T,DOZ,DOZA
A,DO,XA1,DEOR
INITIAL CALC.
C = P/(R*T)
Y1 = A/e^(DEOR/T)
Z = (Y1/DOZA)^½
Z1 = -Z2
DIFF = 0.
NR = 1
NC = 0
XA2 = .10
XB1 = 1.0 - XA1
IF Z1 = 0
A
XB2 = 1.0 - XA2
XBN = (XB2 - XB1)/In(XB2/XB1)
RA = Y1*XA2*C/Z
RA3 = 1-e^(Z*Z1)
RAZ = RA * RA3
XA(NR) = X
YA(NR) = Y
XA2 = Y
NC = 2
FIRST ITERATION ONLY
B
IF CONVERGENCE
XA2N = (C*DOZ*XA1 - Z1*XBN*RAZ)/C*DOZ
X = XA2
Y = XA2N
XR = (XA(NR)*Y - YA(NR)*X)/(XA(NR) - X + Y - YA(NR))
NR = NR + 1, XA(NR) = X, YA(NR) = Y, XA2 = XT

Figure B.8. ACDPRO flow chart

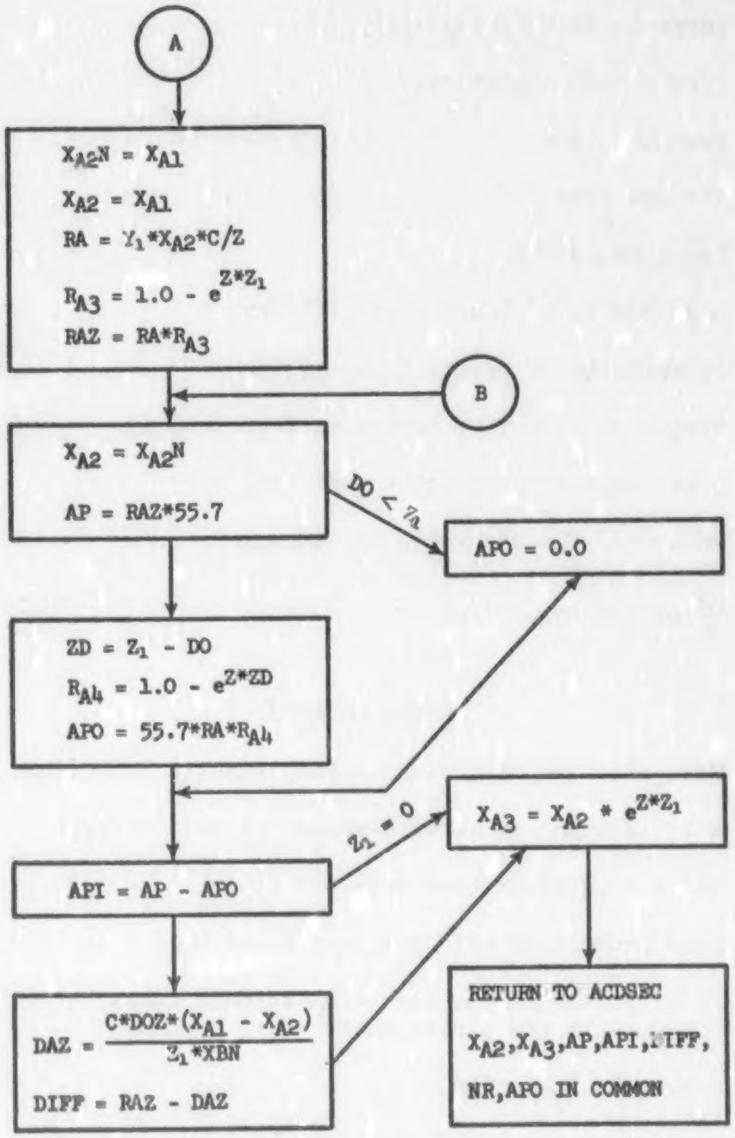

A
$X_{A2}N = X_{A1}$
$X_{A2} = X_{A1}$
$RA = Y_1 * X_{A2} * C/Z$
$R_{A3} = 1.0 - e^{Z*Z_1}$
$RAZ = RA*R_{A3}$
B
$X_{A2} = X_{A2}N$
$AP = RAZ*55.7$
$DO < Z_1$
$APO = 0.0$
$ZD = Z_1 - DO$
$R_{A4} = 1.0 - e^{Z*ZD}$
$APO = 55.7*RA*R_{A4}$
$Z_1 = 0$
$X_{A3} = X_{A2} * e^{Z*Z_1}$
$API = AP - APO$
$DAZ = \frac{C*DOZ*(X_{A1} - X_{A2})}{Z_1*XBN}$
$DIFF = RAZ - DAZ$
RETURN TO ACDSEC
X_{A2},X_{A3},AP,API,DIFF,
NR,APO IN COMMON

Figure B.8. Continued

Table B.14. ACDPRO PROGRAM INPUT AND OUTPUT

ACDPRO INPUTS

Z_1	Inert layer thickness (ft)
Z_2	Pyrite layer thickness
P	Pressure (atm)
R	Gas Law constant
T	Temperature (°R)
DOZ	Diffusivity of inert layer (ft^2/hr)
DOZA	Diffusivity of pyrite layer (ft^2/hr)
A	Frequency factor of Arrhenius form (hr-1)
DO	Depth washed by direct runoff (ft)
X_{A1}	Mole fraction of oxygen in atmosphere
DEOR	$\frac{\Delta E}{R}$ of Arrhenius form

ACDPRO OUTPUTS

X_{A2}	Mole fraction of oxygen at inert-pyrite interface
X_{A3}	Mole fraction at lower boundary of pyrite layer
AP	Total acid production rate per hour (as lb $CaCO_3$)
API	Acid production rate per hour below DO (as lb $CaCO_3$)
DIFF	Final difference between flux through inert layer and oxygen consumed in the pyrite layer
NR	Number of iterations
APO	Acid production rate per hour above DO (as lb $CaCO_3$)

have to be adjusted until the model simulates known data but laboratory tests on soil from the site may provide initial trial values. The errors introduced by the approximate iterative solution can be compensated for and the method is considered adequate and more practical than a more exact (but computer time consuming) method since the model must be calibrated against field data in any case. The use of ACDPRO by CRPSMM will be described in the following section.

CRPSMM PROGRAM DESCRIPTION

The Combined Refuse Pile - Strip Mine Model (CRPSMM) is a modification of the Refuse Pile Model previously presented and can best be described by comparing the CRPSMM with the original program (attached as Refuse Pile Computer Program). CRPSMM consists of four parts: an ACDPRO subroutine described earlier, two other subroutines termed ACDSEC and ACSTR, and the main program of the Refuse Pile Model as modified to accept the three above named subroutines.

ACDSEC, along with ACDPRO, replaces the constant acid production rate used by the Refuse Pile Model. For each acid producing area in the watershed the ACDSEC subroutine inputs the parameters which describe each area to ACDPRO, which calculates the acid production rate for that area only. These individual acid production rates along with the soil volume above the DO line are available to the main program through a common storage. Figure B.9 is a listing of the ACDSEC subroutine.

ACDSTR is a subroutine which, when called, prints for each acid producing area the amount of acid stored in the watershed and the amount of acid removed during the simulation period. There are six storages for acid in the watershed and four flows for acid removal. The subroutine subdivides each of these into substorages and subflows corresponding to the individual acid producing areas. The ten variables output by ACDSTR are described below and have a subscript identification which corresponds to the numbering of the acid producing areas.

AMTACU - acid stored in soil above DO line (lb)
AMTACL - acid stored in soil below DO line (lb)
EXADIR - acid dissolved in direct runoff storage water (lb)
EXAUZ - acid dissolved in upper zone storage water (lb)
EXAINT - acid dissolved in interflow storage water (lb)
EXALZ - acid dissolved in lower zone storage water (lb)
ARDIRT - total acid removed in direct runoff (lb)
ARINFT - total acid removed in interflow (lb)
ARBFLT - total acid removed in baseflow (lb)
ARDSRT - total acid removed to deep storage (lb)

The above variables are all cumulative amounts, updated for each time increment (15 minutes, normally) in the SWM. A listing of the ACDSTR

```
C      ACID PRODUCTION SECTION
C
       SUBRUUTINE ACDSEC
       COMMON/ANM1/APIT,APOT,A1T,A1D,APD(10,14),N
       COMMON/ANM2/Z1,Z2,P,R,1,DOZ,A,DO,XA1,DEOR,DOZA,XA2,XA3,AP,API,DIFF
      *,NR,APO
     1 FORMAT(5F12.6)
     4 FORMAT('0    APOT             APIT             APT')
     8 FORMAT('0 THE HOURLY ACID PRODUCTION CALCULATED IS')
       APIT=0.0
       APOT=0.0
       A1T=0.
       A1D=0.
       DO410I=1,N
       Z1=APD(I,1)
       Z2=APD(I,2)
       DOZ=APD(I,3)
       DOZA=APD(I,4)
       A=APD(I,5)
       DO=APD(I,6)
       A1=APD(I,7)
       CALL ACDPRO
       A1T=A1T+A1
       A1D=A1D+A1*DO
       APOT=APOT+APO*A1
       APD(I,8)=XA2
       APD(I,9)=XA3
       APD(I,10)=AP*A1
       APD(I,11)=APO*A1
       APD(I,12)=API*A1
       APD(I,13)=DO*A1
       APD(I,14)=APO/DO
   410 APIT=APIT+API*A1
       APT=APOT+APIT
       WRITE(6,3)
       WRITE(6,2)
     3 FORMAT('1  ACDSEC OUTPUTS')
     2 FORMAT('0AREA  XA2       XA3        AP         APO        API        VOL
      *     RATE')
     5 FORMAT(I4,F7.3,F7.3,3F10.4,F10.1,F12.8)
       DO420I=1,N
   420 WRITE(6,5)I,(APD(I,J),J=8,14)
       RATE=APOT/A1D
       WRITE(6,6)APT,APOT,APIT,A1D,RATE
     6 FORMAT('0TOTAL WATERSHED    '   ,F7.3,2F10.3,F10.0,F12.7)
       RETURN
       END
```

Figure B.9. The ACDSEC Subroutine Program

subroutine is given in Figure B.10, and the main (CRPSMM) program listing is given at the rear of this appendix. All lines which have been modified or added to the Refuse Pile Model are identified with the initials ANM and a card number on the right hand side in columns 73 to 80. Internal variables are defined in Table B.15. Lines 1 through 57 provide common storage for variables (shared with ACDPRO, ACDSEC, and ACDSTR), initialize default values, and read in and print out the input parameters to the model. These input parameters are defined in Table B.16. Many of the input parameters are subscripted so that each individual acid producing area of the watershed is treated separately. The use of individual calculation of acid production and acid removal for each area of the watershed that differs in inert cover thickness, pyrite thickness, diffusivity in pyrite or inert cover layer, reactivity of pyrite, and the depth leached by direct runoff allows much flexibility of the model. Each area of the watershed that differs significantly from any other area in any of the above six parameters is handled separately. Areas that are not adjacent but are not significantly different should be combined as a single effective area since the model can not differentiate on basis of location within the watershed. The choice of size and numbers of effective acid producing areas must be based on a site inspection and laboratory analysis of parameters such as soil diffusivity, pyrite reactivity, and pyrite location in the soil column. The final choice of areas should be based on the accuracy of simulation required. For some uses one or two areas would be adequate, but in other situations further subdivisions might be made.

Line 58 calls subroutine ACDSEC, which operates as described above. Lines 59 to 63, 74 to 84, and 181 to 194 control the output options and the length of time simulated by the model. Output options are controlled by input parameter NOPT. If NOPT is 0, the program will simulate several years of data from the SWM but will give storm details for any one continuous period of time during each water year, which may be an entire year or any fraction thereof. If NOPT is 1, the program simulates a short time interval; in this case initial conditions of acid stored in the soil and dissolved in the water must be input. The storm details are output but the yearly summaries are not. If NOPT is 2, the program operates the same as when NOPT is 1 but the yearly summaries are also output. They will, of course, be incomplete as the simulation would not normally extend over a full year, but the data are useful for some types of work.

Lines 64 and 65 reads the initial value of acid stored in the watershed at the start of the simulation. The variables are subscripted so that the program reads values for each acid producing area in the watershed.

Line 66 calls subroutine ACDSTR which prints out these initial values.

Lines 67 to 73 read the input from the SWM and change the temperature of the air over the watershed ZTMP from °F to °R as variable T.

```
      SUBROUTINE ACDSTR
      COMMON/ANM1/APIT,APDT,ALT,AID,APD(10,14),N
      COMMON/ANM5/EXADIR(10),EXAINT(10),EXALZ(10),EXAUZ(10),AMTACU(10),
     *AMTACL(10)
      COMMON/ANM6/DDYR1,FA,DAY,J
      COMMON/ANM7/ARDIRT(10),ARINFT(10),ARBFLT(10),ARDSRT(10)
      WRITE(6,1)DDYR1,FA,DAY,J
    1 FORMAT('1 THE ACID STORED AT',I2,'/',I2,'/',I2,'/',I2)
      WRITE(6,3)
    3 FORMAT('0     I   AMTACU     AMTACL     EXADIR      EXAUZ     EXAINT
     * EXALZ')
      DO100I=1,N
  100 WRITE(6,8)I,AMTACU(I),AMTACL(I),EXADIR(I),EXAUZ(I),EXAINT(I),EXALZ
     *(I)
    8 FORMAT(I6,6F10.3)
      WRITE(6,4)
    4 FORMAT('OTOTAL ACID REMOVED')
      WRITE(6,5)
    5 FORMAT('0     I     ARDIRT       ARINFT       ARBFLT       ARDSRT')
      DO110I=1,N
  110 WRITE(6,2)I,ARDIRT(I),ARINFT(I),ARBFLT(I),ARDSRT(I)
    2 FORMAT(I6,4F12.4)
      RETURN
      END
```

Figure B.10. The ACDSTR Subroutine Program

Table B.15. CRPSMM INTERNAL VARIABLES

EXADIR(I)	Weight of acid dissolved in direct runoff storage
EXAINT(I)	Weight of acid dissolved in interflow storage
EXAUZ(I)	Weight of acid dissolved in upper zone storage
EXALZ(I)	Weight of acid dissolved in lower zone storage
AMTACU(I)	Weight of acid adsorbed in upper zone
AMTACL(I)	Weight of acid adsorbed in lower zone
FACDIR(I)	Acid concentration in water entering direct runoff
FACUZ(I)	Acid concentration in water entering the upper zone
FACINT(I)	Acid concentration in water entering interflow
FACLZ(I)	Acid concentration in water entering lower zone
ACDDIR(I)	Acid removed from soil by direct runoff
ACDUZ(I)	Acid removed from soil by water entering upper zone
ACDINT(I)	Acid removed from soil by interflow
ACDLZ(I)	Acid removed from soil by water entering the lower zone
AREDIR(I)	Acid load from direct runoff - lb acidity as $CaCO_3$
AREINF(I)	Acid load from interflow - lb acidity as $CaCO_3$
AREBFL(I)	Acid load from baseflow - lb acidity as $CaCO_3$
AREDSR(I)	Acid routed to deep storage - lb acidity as $CaCO_3$
APD(I,11)	Hourly acid production in upper soil
APD(I,12)	Hourly acid production in lower soil
APD(I,13)	Volume of acid producing soil washed by direct runoff
AA	Ratio of acid producing area to area of watershed
I	Number of acid producing area
ARDIRT(I)	Acid removed by direct runoff during simulation
ARINFT(I)	Acid removed by interflow during simulation
ARBFLT(I)	Acid removed by baseflow during simulation
ARDSRT(I)	Acid removed to deep storage during simulation
ARRDIR	Total acid removed by direct runoff during time interval
ARRINF	Total acid removed by interflow during time interval
ARRBFL	Total acid removed by baseflow during time interval

Table B.16. CRPSMM INPUT VARIABLES

CO(I)	Exponent affecting leaching of acid by direct runoff
CEU(I)	Exponent affecting leaching of acid by water entering the upper zone
OFF(I)	Constant affecting leaching of acid by direct runoff
UZF(I)	Constant affecting leaching of acid by water entering upper zone
IFF(I)	Constant affecting leaching of acid by interflow
LZF(I)	Constant affecting leaching of acid by water entering lower zone
SOLACD	Acid solubility in mg/ℓ
AREA	Total watershed area (ft^2)
FACRED	Adjustment factor for acid entering receiving water by direct runoff
FACRUZ	Adjustment factor for acid entering receiving water by direct runoff from the upper zone
FACREI	Adjustment factor for acid entering receiving water by interflow
FACRLZ	Adjustment factor for acid entering receiving water by interflow from the lower zone
FACREB	Adjustment factor for acid entering receiving water by baseflow
FACRDS	Adjustment factor for acid entering deep storage
FACFD	Factor to adjust direct runoff
FACFI	Factor to adjust interflow
FACFB	Factor to adjust baseflow

Lines 85 to 88 set ARRDIR, ARRINF, ARRBFL, and ARRDSR equal to 0 before the calculations in the fractional hour loop are made. These variables sum the acid removed by direct runoff, interflow, baseflow and that routed to deep storage, respectively, for later use in calculating the yearly summaries. All acid removed from the watershed is included in these totals.

Lines 89 to 174 is the DO loop, which is run once for each acid producing area of the watershed. The operations will be described completely for acid removed by direct runoff and other flows will be described only when handled differently from direct runoff. The loop itself is included in a larger loop which is used once for each set of input data from the SWM. The SWM outputs data on a time interval that corresponds to the approximate time of water flow between isochrones of the basin studied. The CRPSMM time interval must correspond to the SWM.

Line 90 computes AA which is the ratio of the acid producing area to the watershed area. Lines 91 to 92 add the acid produced to the soil storages. The variable TIME is zero if it is raining, so that the model simulates acid production only when the watershed is not receiving precipitation. The single soil storage used in the Refuse Pile Model (AMTACD) is divided into that part leached by direct runoff (AMTACU) and that part not leached by direct runoff (AMTACL). Definition of these storages conform to the ACDPRO subroutine; i.e., AMTACU is above depth DO and AMTACL below DO.

In the Refuse Pile Model the acid removed by water entering direct runoff storage is simulated by

ACDDIR = FACDIR*SOLACD*OVFLST*CF

where ACDDIR = the acid removed in 15 minutes in lb acidity as $CaCO_3$
FACDIR = an adjustment factor,
SOLACD = the assumed value of acid solubility in lb/ft^3,
OVFLST = the water entering direct runoff storage in inches of precipitation, and
CF = a conversion factor to convert inches of precipitation to ft^3

Essentially this definition of ACDDIR calls for the removal of acid at a constant concentration determined by SOLACD and FACDIR.

In CRPSMM the definition of FACDIR is altered in accordance with the equation in line 94

FACDIR(I) = OFF(I)*(AMTACU(I)/APD(I,13))/(OVFLST*APD(I,7)/AREA)**CO(I)

where FACDIR(I) = the concentration of acid going to direct runoff storage from area I in lb/ft^3,
OFF(I) = an input parameter for area I,
AMTACU(I) = the acid stored in the upper soil in area I (lb),
APD(I,13) = the volume of upper soil in area I (ft^3),
OVFLST = as defined above,
APD(I,7) = the area of area I (ft^2),
AREA = the total watershed area (ft^2), and
CO(I) = an input parameter for area I.

This equation was developed for two reasons: First, the New Kathleen Refuse Pile[4] data indicated that the log acidity and log flow rate followed an approximately linear relationship. This is an equation of the form

$$Y = K/X^n$$

where Y is acidity and X the flow rate. Secondly, both fundamental mass transfer considerations and the limited field data observed require that the acid in direct runoff decrease as the amount of acid in the soil leached by direct runoff decrease. To reflect this the constant of proportionality k is replaced by OFF(I)*(AMTACU(I)/APD(I,13)). The ratio AMTACU(I)/APD(I,13) is an approximation of the "effective concentration" of acid adsorbed on the soil in the upper layer. Both OFF(I) and CO(I) are input adjustment factors used to fit the model to actual data. Line 95 tests FACDIR(I) to see if it is greater than SOLACD. If so, it is replaced by SOLACD, the limiting solubility input to the model. The defining equation for ACDDIR(I) line 103 then becomes

$$ACDDIR(I) = FACDIR(I)*OVFLST*AA*DF$$

where ACDDIR(I) is the acid going to direct runoff storage from area I,
FACDIR(I) replaces both FACDIR and SOLACD from the Refuse Pile Model, and the variables are as described above, and
AA adjusts the amount of acid removed so that it includes the precipitation on the specific acid producing area only.

In lines 99 and 104 FACUZ(I), the concentration of acid entering the upper zone, and ACDUZ(I), the amount of acid entering the upper zone, are determined in an analogous way. However, different input parameters are used. FACINT(I), the concentration of acid entering interflow storage, and FACLZ(I), the concentration of acid entering the lower zone, are defined as a fraction of the maximum solubility (SOLACD) by input parameters IFF(I) and LZF(I), respectively, in lines 105 and

107. The determination of the amount of acid removed by these two flows ACDINT(I) and ACDLZ(I) are also analogous to ACDDIR(I).

The acid to be removed from the soil is subtracted from the layer above DO (that washed by direct runoff) AMTACU(I) in line 111 is there is sufficient acid there. If not, the acid removed by water going to interflow storage and lower zone storage are removed from the lower soil layer. AMTACL(I), in line 113, and the acid removed by direct runoff and water going to the upper zone are set at zero, since these flows can not contact the lower soil layer by definition. If there is no acid in the lower soil storage, then all acid removal flows are set equal to zero. The assumption is made in the model that acid will be removed preferentially from the upper soil since this is the layer leached first by incoming precipitation.

The acid removed from the soil is stored as a water solution in four places; the direct runoff storage (EXADIR(I)), the upper zone (EXAUZ(I)), interflow storage (EXAINT(I)), and the lower zone (EXALZ(I)).

It is assumed that the ratio of weight of acid in direct runoff reaching the measuring point (AREDIR(I)), to the acid stored in direct runoff storage EXADIR(I) is proportional to the ratio of direct runoff to direct runoff storage. This is accomplished by line 128,

```
AREDIR(I) = (DIRRNF/OVLDST)*EXADIR(I)*FACRED*AA
```

where DIRRNF = the direct runoff entering the stream in inches of precipitation,
OVLDST = the direct runoff storage in inches of precipitation, and
FACRED = an input adjustment factor and the other variables are as defined above.

Again the concentration is tested in line 130 to ensure it does not exceed SOLACD.

The acid removed from the other three storages EXAUZ(I), EXAINT(I), and EXALZ(I) is calculated in an analogous manner. This method of calculation assumes that the water and acid in each storage are completely mixed. The paths of acid flow from the four water storages are the same as in the Refuse Pile Model. A block diagram of water and acid flow is given in Figure B.11. Acid removed by direct runoff comes first from EXADIR(I), then from EXAUZ(I) if EXADIR(I) is depleted. If direct runoff stops before EXAUZ(I) is exhausted the remainder is added to AMTACU(I) in line 139. This acid is mostly in depression storage from which the water will evaporate, thus precipitating the acid back to the upper soil storage AMTACU(I). EXAINT(I) can only be depleted by interflow. If this storage is exhausted the acid in interflow will be drawn

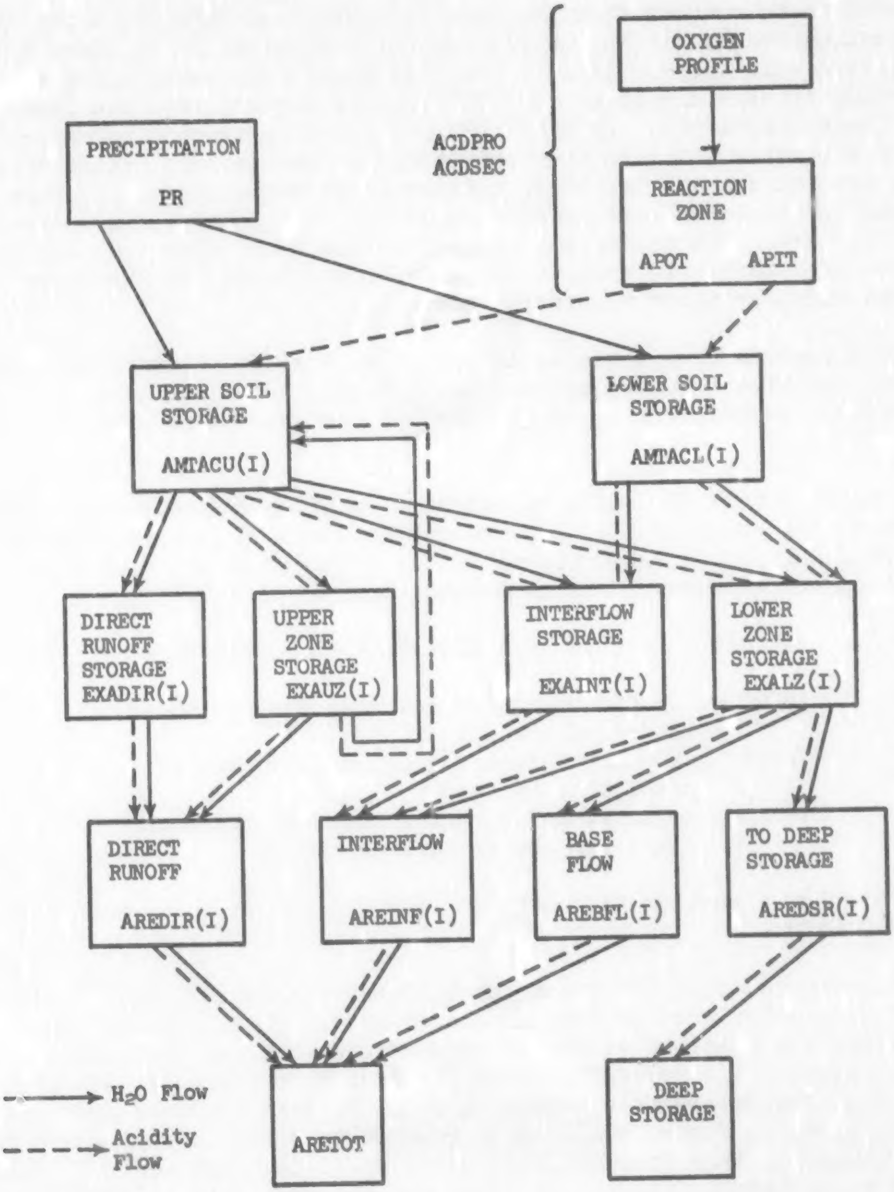

OXYGEN PROFILE
ACDPRO
ACDSEC
REACTION ZONE
APOT
APIT
PRECIPITATION
PR
UPPER SOIL STORAGE
AMTACU(I)
LOWER SOIL STORAGE
AMTACL(I)
DIRECT RUNOFF STORAGE
EXADIR(I)
UPPER ZONE STORAGE
EXAUZ(I)
INTERFLOW STORAGE
EXAINT(I)
LOWER ZONE STORAGE
EXALZ(I)
DIRECT RUNOFF
AREDIR(I)
INTERFLOW
AREINF(I)
BASE FLOW
AREBFL(I)
TO DEEP STORAGE
AREDSR(I)
ARETOT
DEEP STORAGE
H2O Flow
Acidity Flow

Figure B.11. CRPSMM (Combined Refuse Pile - Strip Mine Model) block diagram

from the acid stored in the lower zone, EXALZ(I). The acid components removed by baseflow and routed to deep storage are also drawn from the lower zone.

Three variables, UZSN, LZSN, and ZTMP(I,J), which are obtained from the SWM, are not used in the present form of CRPSSM. These have been retained, however, to provide for future modifications of the model. There is a possibility that the diffusivity of the soil can be correlated with UZSN or LZSN and that ZTMP(I,J) can be used to define the soil temperature if it is found that the acid production rate should be calculated on a monthly or daily basis due to temperature changes. The latter modification can be made simply by calling ACDSEC when the variable DAY = 1 if a monthly calculation is desired or when J = 1 if a daily calculation is desired. Another possibility to account for temperature variable acid production rate is to input and use measured average monthly soil temperatures and call ACDSEC each month.

COMBINED REFUSE PILE - STRIP MINE MODEL (CRPSMM) COMPUTER PROGRAM

```
C
C         COMBINED REFUSE PILE - STRIP MINE MODEL
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      COMMON/ANM1/ARIT,ARDI,AIT,AID,ARD(10,14),N                       ANM   1
      COMMON/ANM2/Z1,Z2,P,R,I,DOZ,A,DO,XA1,DEOR,DOZA,XA2,XA3,AP,API,DIFF
     *,NR,APO                                                           ANM   2
      COMMON/ANM3/FACDIR(10),FACUZ(10),FACINT(10),FACLZ(10)            ANM   3
      COMMON/ANM4/CO(10),CEU(10),OFF(10),IFF(10),UZF(10),LZF(10)       ANM   4
      COMMON/ANM5/EXADIR(10),EXAINT(10),EXALZ(10),EXAUZ(10),AMTACU(10),
     *AMTACL(10)                                                        ANM   5
      COMMON/ANM6/DDYR1,FA,DAY,J                                        ANM   6
      COMMON/ANM7/ARDIRT(10),ARINET(10),ARBELT(10),ARDSRT(10)          ANM   7
      REAL LZS,LZSN,INTF,IFF,LZF                                        ANM   8
      INTEGER YEAR1,MONTH1,DAY1,YEAR2,MONTH2,DAY2,YEARED,DDYR1,FA,DAY,J,
     1DD23,MONED,YEARPR,YEARST,COUNT,DDAY,DAYOUT,DAYEND
      DIMENSION ARD(12,31),ARI(12,31),ARB(12,31),SAD(12),SAI(12),SAB(12)
     1,TACD(12,31),STACD(12),TSTR(12,31),STSTR(12),SDRR(12,31),SSDRR(12)
     2,SINT(12,31),SSINT(12),SBAS(12,31),SSBAS(12),DDAY(32)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      DIMENSION ACDDIR(10),ACDUZ(10),ACDINT(10),ACDLZ(10)              ANM   9
      DIMENSION AREDIR(10),AREINE(10),AREBEL(10),AREDSR(10)            ANM  10
C     FORMAT SECTION
C
    1 FORMAT(6F12.6)                                                    ANM  11
    2 FORMAT('1 THE INPUT DATA IS')                                     ANM  12
  403 FORMAT('0   I     Z1        Z2        DOZ      DOZA       A
     *  DO        AI')                                                  ANM  13
  401 FORMAT(2I10)                                                      ANM  14
  402 FORMAT(4F10.5,E12.4,F10.5,F12.0)                                  ANM  15
    4 FORMAT(3F12.6,E12.4,F12.6)                                        ANM  16
  404 FORMAT(I5,4F10.5,F12.5,F10.5,F12.0)                               ANM  18
  240 FORMAT(6F12.1)                                                    ANM  18
    3 FORMAT('0       P          R          XA1        DEOR   ')        ANM  19
    5 FORMAT('0      CO         CEU         OFF        UZF         IFF
     *         LZF')                                                    ANM  20
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      INPUT DATA
    6 FORMAT('0                SOLACD                 AREA')             ANM  21
    7 FORMAT('0 FACRED  FACRUZ  FACREI  FACRLZ  FACREB  FACRDS  FACFD
     *FACFI   FACFB')                                                   ANM  22
    9 FORMAT('0      XA2         XA3          APO         API        AP') ANM  23
 2000 FORMAT(4I10)                                                      ANM  24
 2001 FORMAT(2F20.2)                                                    ANM  25
 2002 FORMAT(9F8.4)                                                     ANM  26
      WRITE(6,2)                                                        ANM  27
      READ(5,4)P,R,XA1,DEOR                                             ANM  28
      WRITE(6,3)                                                        ANM  29
      WRITE(6,4)P,R,XA1,DEOR                                            ANM  30
      READ(5,401)N,NOPT                                                 ANM  31
C          N=NO. OF DIFFERING ACID PRODUCING AREAS
C          IF NOPT=0 PROGRAM RUNS SEVERAL YRS. OF DATA WITH DETAILED OUTPUT
C     FOR ONE TIME PERIOD PER YR. *** IF NOPT=1 PROGRAM RUNS FOR SELECTED
C     PERIOD ONLY WITHOUT MONTHLY OR YEARLY SUMMARIES *** IF NOPT=2 PROGRAM
C     RUN IS SAME AS NOPT=1 BUT WITH SUMMARIES
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C INITIALING THE AMOUNT OF ACID IN THE FOUR ZONES PLUS THE  TOTAL
C        AMOUNT OF ACID IN THE PILE - DEFAULT VALUES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
      K=0                                                               ANM 32
      T=530.                                                            ANM 33
      DO101I=1,N                                                        ANM 34
      ARDIRT(I)=0.                                                      ANM 35
      ARINFT(I)=0.                                                      ANM 36
      ARBELT(I)=0.                                                      ANM 37
      ARDSRT(I)=0.                                                      ANM 38
      EXADIR(I)=0.                                                      ANM 39
      EXAINT(I)=0.                                                      ANM 40
      EXALZ(I)=0.                                                       ANM 41
      EXAUZ(I)=0.                                                       ANM 42
      AMTACU(I)=0.                                                      ANM 43
  101 AMTACL(I)=0.                                                      ANM 44
      WRITE(6,403)                                                      ANM 45
      DO 400I=1,N                                                       ANM 46
      READ(5,402)APD(I,1),APD(I,2),APD(I,3),APD(I,4),APD(I,5),APD(I,6),A
     *PD(I,7)                                                           ANM 47
  400 WRITE(6,404)I,(APD(I,J),J=1,7)                                    ANM 48
      WRITE(6,5)                                                        ANM 49
      DO410I=1,N                                                        ANM 50
      READ(5,1)CO(I),CEU(I),OFF(I),UZF(I),IFF(I),LZF(I)                 ANM 51
  410 WRITE(6,1)CO(I),CEU(I),OFF(I),UZF(I),IFF(I),LZF(I)                ANM 52
      READ(5,2001)SOLACD,AREA                                           ANM 53
      WRITE(6,6)                                                        ANM 54
      WRITE(6,2001)SOLACD,AREA                                          ANM 55
      READ (5,2002)FACRED,FACRUZ,FACREI,FACRLZ,FACREB,FACRDS,FACFD,FACFI
     *,FACFB
      WRITE(6,7)                                                        ANM 56
      WRITE(6,2002)FACRED,FACRUZ,FACREI,FACRLZ,FACREB,FACRDS,FACFD,FACFI
     *,FACFB                                                            ANM 57
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CALL ACDSEC                                                       ANM 58
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C CHANGING SOLACD TO  POUNDS PER CUBIC FEET
C FACTOR TO CHANGE INCHES OF RUNOFF OF REFUSE PILE TO CUBIC FEET PER
C        SECOND OF RUNOFF
C FACTOR TO CHANGE INCHES OF RUNOFF FROM REFUSE PILE TO CUBIC FEET OF
C        RUNOFF (BASED ON 15 MINUTE INTERVAL)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SOLACD= SOLACD*2.205/(35310)
      CFS=AREA*0.24*0.268858*0.0003587
      CF=CFS*60.*15.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C THE NEXT LEAP YEAR WILL OCCUR IN WHAT WATER YEAR -  LAST TWO DIGITS
C        DIVIDED BY 4
C THE STARTING WATER YEAR OF THE INPUT DATA FROM THE STANFORD WATERSHED
C        MODEL
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      YEARLP=14.75
      YEARST=58
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C READING IN DATA FOR SPECIFIC DAY YOU WANT OUTPUT - SET FOR ONE TIME
C        PER WATER YEAR.  YEAR1= WATER YEAR, MONTH1 = MONTH, DAY1 = DAY
C        TO START OUTPUT, NDAY = NUMBER OF CONSECTIVE DAYS OF OUTPUT
C        REQUESTED
C        THE TIME IS BASED ON A WATER YEAR WHERE THE FIRST MONTH IS
C        OCTOBER
```

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 9100 READ(5,2000) YEAR1,MONTH1,DAY1,NDAY
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHECKING IF ENOUGH DATA HAS BEEN OUTPUTED AS REQUESTED:  IF SO
C        WILL STOP
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(YEAR1 .EQ. 0) GO TO 9200
      DAYEND=NDAY*96
      IF(K.EQ.1)GO TO 530                                             ANM 59
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  FOR EACH WATER YEAR INITILIZE THE DAILY, MONTHLY, AND YEARLY VALUES.
C        SET EVERY THING EQUAL TO ZERO
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  560 DAYOUT=0                                                        ANM 60
 8900 DDAY(1)=1.
      DO 200 I=1,12
      DO 201 J=1,31
      ARD(I,J)=0.
      ARI(I,J)=0.
      DDAY(J+1)=DDAY(J)+1.
      TACD(I,J)=0.
      TSTR(I,J)=0.
      SDRR(I,J)=0.
      SINT(I,J)=0.
      SBAS(I,J)=0.
  201 ARB(I,J)=0.
      SAD(I)=0.
      SAI(I)=0.
      STACD(I)=0.
      STSTR(I)=0.
      SSDRR(I)=0.
      SSINT(I)=0.
      SSBAS(I)=0.
  200 SAB(I)=0.
      SUMAD=0.
      SUMAI=0.
      SUMAB=0.
      SUMTA=0.
      SUMST=0.
      SUMSB=0.
      SUMSI=0.
      SUMSD=0.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  PROCEDURE SO THAT THE EXTRA DAYS IN THE MONTH WILL CREATE AN OVER-
C        FLOW IN THE OUTPUT DATA.  WILL ADJUST FOR THE LEAP YEAR.
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SLYEAR=FLOAT(YEARST)/4.
      COUNT=0.
      J=2
 7000 I=31
      ARI(J,I)=100000000.
      TACD(J,I)=100000000.
 7001 ARD(J,I)=100000000.
      ARB(J,I)=100000000.
      TSTR(J,I)=100000000.
      SDRR(J,I)=100000000.
      SINT(J,I)=100000000.
      SBAS(J,I)=100000000.
      COUNT=COUNT+1.
      IF(COUNT .EQ. 1.) J=7
      IF(COUNT .EQ. 3.) J=12
      IF(COUNT .EQ. 2.) J=9
      IF(COUNT .EQ. 4.) J=5
      IF(COUNT .EQ. 5.) GO TO 7004
      IF(COUNT .EQ. 6. .AND. SLYEAR .EQ. YEARLP) GO TO 7002
      IF (COUNT .EQ. 6. ) GO TO 7003
      IF(COUNT .EQ. 7.) GO TO 7005
      GO TO 7000
 7004 J=30
      GO TO 7001
 7002 YEARLP=YEARLP+1.
      GO TO 7005
 7003 J=29
      GO TO 7001
 7005 CONTINUE
      K=1                                                          ANM 61
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         INPUT FROM STANFORD WATERSHED MODEL
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  530 IF(NOPT.LE.0)GO TO 8810                                      ANM 62
 8810 KCOUNT=0                                                     ANM 63
C         INPUT STARTING VALUES OF ACID STORAGE
      DO420I=1,N                                                   ANM 64
  420 READ(5,240)AMTACU(I),AMTACL(I),EXADIR(I),EXAUZ(I),EXAINT(I),EXALZ(
     *I)                                                           ANM 65
      CALL ACDSTR                                                  ANM 66
 8800 READ (5,129)DDYR1,FA,DAY,J,DD23,PR,ENTRUZ,ENTRLZ,RGX,OVFLST  ANM 67
      READ (5,134)UZS,LZS,SRGX,OVLDST,UZSN,LZSN,ZTMP               ANM 68
      READ (5,135)DIRRNF,INTF,BASFLW,TOTFLW,OUTFLW,SFX             ANM 69
  129 FORMAT(I6,4I2,5E10.4)                                        ANM 70
  134 FORMAT(7F10.6)                                               ANM 71
  135 FORMAT(6F10.6)                                               ANM 72
      T=ZTMP+460.                                                  ANM 73
      IF(NOPT.LE.0)GO TO 71                                        ANM 74
      IF(KCOUNT.GT.1)GO TO 71                                      ANM 75
      IF(DDYR1.EQ.YEAR1)GO TO 210                                  ANM 76
      GO TO 8800                                                   ANM 77
  210 IF(FA.EQ.MONTH1)GO TO 220                                    ANM 78
      GO TO 8800                                                   ANM 79
  220 IF(DAY.EQ.DAY1)GO TO 230                                     ANM 80
      GO TO 8800                                                   ANM 81
  230 CONTINUE                                                     ANM 82
      KCOUNT=KCOUNT+1                                              ANM 83
   71 CONTINUE                                                     ANM 84
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C         THE TIME INTERVAL MUST CORRESPOND WITH THE TIME
C       INTERVAL FROM THE STANFORD WATERSHED MODEL
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      TIME=.25
      IF(PR .NE. 0.0) TIME=0.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CALCULATING THE AMOUNT OF ACID BEING PRODUCED
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      ARRDIR=0.                                                    ANM 85
      ARRINF=0.                                                    ANM 86
      ARRRFL=0.
      ARRDSR=0.
      DO430I=1,N
      AA=APD(I,7)/AREA
      AMTACU(I)=APD(I,11)*TIME+AMTACU(I)
      AMTACL(I)=APD(I,12)*TIME+AMTACL(I)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CALCULATING THE AMOUNT OF ACID REMOVED
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      IF(OVFLST.LE.0.0)GO TO 80
      FACDIR(I)=OFF(I)*(AMTACU(I)/APD(I,13))/(OVFLST*APD(I,7)/AREA)**CO(
     *I)
      IF(FACDIR(I).GT.SOLACD)FACDIR(I)=SOLACD
      GO TO 82
   80 FACDIR(I)=0.
   82 IF(ENTRUZ.LE.0.0) GO TO 81
      FACUZ(I)=UZF(I)*(AMTACU(I)/APD(I,13))/(ENTRUZ*APD(I,7)/AREA)**CEU(
     *I)
      IF(FACUZ(I).GT.SOLACD)FACUZ(I)=SOLACD
      GO TO 83
   81 FACUZ(I)=0.
   83 ACDDIR(I)=FACDIR(I)*OVFLST*AA*CF
      ACDUZ(I)=FACUZ(I)*ENTRUZ*AA*CF
      FACINT(I)=IFF(I)*SOLACD
      IF(FACINT(I).GT.SOLACD)FACINT(I)=SOLACD
      FACLZ(I)=LZF(I)*SOLACD
      IF(FACLZ(I).GT.SOLACD)FACLZ(I)=SOLACD
      ACDINT(I)=FACINT(I)*RGX*AA*CF
      ACDLZ(I)=FACLZ(I)*ENTRLZ*AA*CF
C     TOTALS FOR UPPER ZONE AND LOWER ZONE
      AMTACU(I)=AMTACU(I)-ACDDIR(I)-ACDINT(I)-ACDUZ(I)-ACDLZ(I)
      IF(AMTACU(I))58,58,59
   58 AMTACL(I)=AMTACL(I)-ACDINT(I)-ACDLZ(I)
      AMTACU(I)=0.
      ACDDIR(I)=0.
      ACDUZ(I)=0.
      IF(AMTACL(I))69,69,57
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  STORING ACID IN FOUR ZONES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   59 EXADIR(I)=ACDDIR(I)+EXADIR(I)
      EXAUZ(I)=ACDUZ(I)+EXAUZ(I)
   57 EXAINT(I)=ACDINT(I)+EXAINT(I)
      EXALZ(I)=ACDLZ(I)+EXALZ(I)
      GO TO 60
   69 AMTACL(I)=0.
      ACDINT(I)=0.
      ACDLZ(I)=0.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  REMOVING THE ACID
C  CHECKING TO MAKE SURE THERE IS ACID TO BE REMOVED
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
   60 IF(DIRRNF .EQ. 0.0) GO TO 22
      IF(EXADIR(I).LE.0.)GO TO 21
      IF(OVLDST.LE.0.)GO TO 21
      AREDIR(I)=(DIRRNF/LVLDST)*EXADIR(I)*FACRED*AA
      IF(EXADIR(I).LE.AREDIR(I))AREDIR(I)=EXADIR(I)
      IF(AREDIR(I).GT.SOLACD*DIRRNF*CF*AA) AREDIR(I)=SOLACD*DIRRNF*CF*AA
      EXADIR(I)=EXADIR(I)-AREDIR(I)
      GO TO 23
   21 IF(EXAUZ(I).LE.0.)GO TO 22                                        ANM132
      IF(UZS.LE.0.)GO TO 22                                             ANM133
      AREDIR(I)=(DIRRNF/UZS)*EXAUZ(I)*FACRUZ*AA                         ANM134
      IF(EXAUZ(I).LE.AREDIR(I))AREDIR(I)=EXAUZ(I)                       ANM135
      IF(AREDIR(I).GT.SOLACD*DIRRNF*CF*AA)AREDIR(I)=SOLACD*DIRRNF*CF*AA ANM136
      EXAUZ(I)=EXAUZ(I)-AREDIR(I)                                       ANM137
      GO TO 23
   22 AREDIR(I)=0.                                                      ANM138
   61 AMTACU(I)=AMTACU(I)+EXAUZ(I)                                      ANM139
      EXAUZ(I)=0.                                                       ANM140
   23 IF(SRGX .EQ. 0.0) GO TO 31                                        ANM141
      AREINF(I)=(INTF/SRGX)*EXAINT(I)*FACREI*AA                         ANM142
      IF(EXAINT(I).LE.0.)GO TO 31                                       ANM143
      IF(EXAINT(I).LE.AREINF(I))AREINF(I)=EXAINT(I)                     ANM144
      IF(AREINF(I).GT.SOLACD*INTF*CF*AA)AREINF(I)=SOLACD*INTF*CF*AA     ANM145
      EXAINT(I)=EXAINT(I)-AREINF(I)                                     ANM146
      GO TO 30
   31 IF(LZS.LE.0.)GO TO 32                                             ANM147
      AREINF(I)=(INTF/LZS)*EXALZ(I)*FACRLZ*AA                           ANM148
      IF(EXALZ(I).LE.AREINF(I))AREINF(I)=EXALZ(I)                       ANM149
      IF(AREINF(I).GT.SOLACD*INTF*CF*AA)AREINF(I)=SOLACD*INTF*CF*AA     ANM150
      EXALZ(I)=EXALZ(I)-AREINF(I)                                       ANM151
      GO TO 30
   32 AREINF(I)=0                                                       ANM152
   30 IF(BASFLW .EQ. 0.0) GO TO 43
      IF(LZS.LE.0.)GO TO 43                                             ANM153
      AREBFL(I)=(BASFLW/LZS)*EXALZ(I)*FACREB*AA                         ANM154
      IF(EXALZ(I).LE.0.) GO TO 43                                       ANM155
      IF(EXALZ(I).LE.AREBFL(I))AREBFL(I)=EXALZ(I)                       ANM156
      IF(AREBFL(I).GT.SOLACD*BASFLW*CF*AA)AREBFL(I)=SOLACD*BASFLW*CF*AA ANM157
      EXALZ(I)=EXALZ(I)-AREBFL(I)                                       ANM158
      GO TO 40
   43 AREBFL(I)=0.                                                      ANM159
   40 IF(INTF .EQ. 0.0) GO TO 51
      IF(LZS.LE.0.)GO TO 52                                             ANM160
      AREDSR(I)=(INTF/LZS)*EXALZ(I)*FACROS*AA                           ANM161
      IF(EXALZ(I).LE.0.)GOTO 52                                         ANM162
      IF(EXALZ(I).LE.AREDSR(I))AREDSR(I)=EXALZ(I)                       ANM163
      IF(AREDSR(I).GT.SOLACD*INTF*CF*AA)AREDSR(I)=SOLACD*INTF*CF*AA     ANM164
      EXALZ(I)=EXALZ(I)-AREDSR(I)                                       ANM165
      GO TO 51
   52 AREDSR(I)=0.                                                      ANM166
C
   51 CONTINUE                                                          ANM167
      ARRDIR=ARRDIR+AREDIR(I)                                           ANM168
      ARRINF=ARRINF+AREINF(I)                                           ANM169
      ARDIRT(I)=ARDIRT(I)+AREDIR(I)                                     ANM170
      ARINFT(I)=ARINFT(I)+AREINF(I)                                     ANM171
      ARBFLT(I)=ARBFLT(I)+AREBFL(I)                                     ANM172
      ARDSRT(I)=ARDSRT(I)+AREDSR(I)                                     ANM173
  430 ARRBFL=ARRBFL+AREBFL(I)                                           ANM174
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C FINDING THE DAILY VALUES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CFDIR=DIRRNF*CFS*FACED
      CFINT=INTF*CFS*FACFI
      CFBAS=BASFLW*CFS*FACEB
      CFTOT=CFBAS+CFINT+CFDIR
      ARETOT=ARRDIR+ARRINF+ARRBFL                                       ANM175
      ARD(FA,DAY)=ARRDIR+ARD(FA,DAY)                                    ANM176
      ARI(FA,DAY)=ARRINF+ARI(FA,DAY)                                    ANM177
      ARB(FA,DAY)=ARRBFL+ARB(FA,DAY)                                    ANM178
      SBAS(FA,DAY)=CFBAS+SBAS(FA,DAY)
      SINT(FA,DAY)=CFINT+SINT(FA,DAY)
      SDRR(FA,DAY)=CFDIR+SDRR(FA,DAY)
      TSTR(FA,DAY)=CFTOT+TSTR(FA,DAY)
      TACD(FA,DAY)=ARRDIR+ARRINF+ARRBFL+TACD(FA,DAY)                    ANM179
   65 IF(YEAR1 .EQ. DDYR1) GO TO 120
      GO TO 100
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHECKING IF THIS IS SPECIFIC DAY TO BE OUTPUTED
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  120 IF(MONTH1 .EQ. FA .OR. DAYOUT .GT. 1) GO TO 130
      GO TO 100
  130 IF(DAY1 .EQ. DAY .OR. DAYOUT .GT. 1) GO TO 131
      GO TO 100
C
  131 IF(DAYOUT.EQ.0)GO TO 132                                          ANM180
      GO TO 133                                                         ANM181
  132 WRITE(6,3030)
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  CHECKING IF THIS IS THE END OF THE SPECIFIC DAY OUTPUT
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
  133 DAYOUT=DAYOUT+1                                                   ANM182
 3030 FORMAT('1',94H YR MO DY HR PD    AREDIR    AREINF    AREBFL    ARE
     1TOT    CFDIR    CFINT    CFBAS    CFTOT)
      WRITE(6,3031) DDYR1,FA,DAY,J,DD23,ARRDIR,ARRINF,ARRBFL,ARETOT,CFD
     1IR,CFINT,CFBAS,CFTOT                                              ANM183
 3031 FORMAT(' ',5(1X,I2),8(1X,F9.4))
      IF(DAYOUT.EQ.DAYEND)GO TO 500                                     ANM184
  100 YEARPR=YEARST+1
      IF(YEARPR .EQ. DDYR1 .AND. FA .EQ. 1) GO TO 8000
      GO TO 8800
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C  FINDING THE MONTHLY VALUES
C  FINDING YEARLY VALUES
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 8000 DO 700 I=1,12
      DO 701 J=1,31
      IF(ARD(I,J) .EQ. 100000000.) GO TU 702
      SAD(I)=ARD(I,J)+SAD(I)
      SAI(I)=ARI(I,J)+SAI(I)
      SAB(I)=ARB(I,J)+SAB(I)
      SSBAS(I)=SBAS(I,J)+SSBAS(I)
      SSINT(I)=SINT(I,J)+SSINT(I)
      SSDRR(I)=SDRR(I,J)+SSDRR(I)
      STSTR(I)=TSTR(I,J)+STSTR(I)
      STACD(I)=TACD(I,J)+STACD(I)
  702 CONTINUE
  701 CONTINUE
      SUMAD=SAD(I)+SUMAD
      SUMAI=SAI(I)+SUMAI
      SUMAB=SAB(I)+SUMAB
      SUMTA=STACD(I)+SUMTA
      SUMST=STSTR(I)+SUMST
      SUMSB=SSBAS(I)+SUMSB
      SUMSI=SSINT(I)+SUMSI
      SUMSD=SSDRR(I)+SUMSD
  700 CONTINUE
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

```
C  OUTPUTING THE '8' DAILY TABLES PLUS THE YEARLY SUMMARY
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      CALL ACDSTR                                                       ANM185
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3003)
 3003 FORMAT(' ',38X,48HSYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS
     1)
      WRITE(6,3013)
      DO 300 I=1,31
      WRITE(6,3011) DDAY(I),(ARD(J,I),J=1,12)
  300 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3002)
 3002 FORMAT(' ',39X,44HSYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS)
      WRITE(6,3013)
      DO 301 I=1,31
      WRITE(6,3011) DDAY(I),(ARI(J,I),J=1,12)
  301 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3001)
 3001 FORMAT(' ',39X,43HSYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS)
      WRITE(6,3013)
      DO 302 I=1,31
      WRITE(6,3011) DDAY(I),(ARB(J,I),J=1,12)
  302 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3004)
 3004 FORMAT(' ',42X,37HSYNTHESIZED TOTAL ACID LOAD IN POUNDS)
      WRITE(6,3013)
      DO 303 I=1,31
      WRITE(6,3011) DDAY(I),(TACD(J,I),J=1,12)
  303 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3006)
 3006 FORMAT(' ',35X,50HSYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECO
     1ND)
      WRITE(6,3013)
      DO 305 I=1,31
      WRITE(6,3020) DDAY(I),(SDRR(J,I),J=1,12)
  305 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3007)
 3007 FORMAT(' ',37X,46HSYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 306 I=1,31
      WRITE(6,3020) DDAY(I),(SINT(J,I),J=1,12)
  306 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3008)
 3008 FORMAT(' ',37X,45HSYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 307 I=1,31
      WRITE(6,3020) DDAY(I),(SBAS(J,I),J=1,12)
  307 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,3005)
 3005 FORMAT(' ',37X,47HSYNTHESIZED STREAMFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,3013)
      DO 304 I=1,31
      WRITE(6,3020) DDAY(I),(TSTR(J,I),J=1,12)
  304 CONTINUE
      WRITE(6,3012) YEARST,YEARPR
      WRITE(6,4005)
 4005 FORMAT('-',3X,3HOCT,6X,3HNOV,6X,3HDEC,6X,3HJAN,6X,3HFEB,6X,3HMAR,6
     1X,3HAPR,6X,3HMAY,6X,3HJUN,6X,3HJUL,6X,3HAUG,6X,3HSEP,3X,10HYEAR TO
     2TAL)
      WRITE(6,4000)
      WRITE(6,4000)
      WRITE(6,4001)
 4001 FORMAT('0',48HSYNTHESIZED ACID LOAD IN DIRECT RUNOFF IN POUNDS)
      WRITE(6,4002) (SAD(I),I=1,12),SUMAD
      WRITE(6,4004)
 4004 FORMAT('0',44HSYNTHESIZED ACID LOAD IN INTERFLOW IN POUNDS)
      WRITE(6,4002) (SAI(I),I=1,12),SUMAI
      WRITE(6,4006)
 4006 FORMAT('0',43HSYNTHESIZED ACID LOAD IN BASEFLOW IN POUNDS)
      WRITE(6,4002) (SAB(I),I=1,12),SUMAB
      WRITE(6,4007)
 4007 FORMAT('0',37HSYNTHESIZED TOTAL ACID LOAD IN POUNDS)
      WRITE(6,4002) (STACD(I),I=1,12),SUMTA
      WRITE(6,4008)
 4008 FORMAT('0',50HSYNTHESIZED DIRECT RUNOFF IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (SSDRR(I),I=1,12),SUMSD
      WRITE(6,4010)
 4010 FORMAT('0',46HSYNTHESIZED INTERFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (SSINT(I),I=1,12),SUMSI
      WRITE(6,4011)
 4011 FORMAT('0',45HSYNTHESIZED BASEFLOW IN CUBIC FEET PER SECOND)
      WRITE(6,4009). (SSBAS(I),I=1,12),SUMSB
      WRITE(6,4012)
 4012 FORMAT('0',78HSYNTHESIZED TOTAL AMOUNT OF WATER ENTERING THE STREA
     1M IN CUBIC FEET PER SECOND)
      WRITE(6,4009) (STSTR(I),I=1,12),SUMST
 3011 FORMAT(' ',3X,I2,12(1X,F9.1))
 3012 FORMAT('1',40X,32HANNUAL SUMMARY FOR WATER YEAR 19,I2,1X,1H-,1X,
     12H19,I2)
 3013 FORMAT('-',3X,3HDAY,5X,3HOCT,6X,3HNOV,6X,3HDEC,6X,3HJAN,6X,3HFEB,6
     1X,3HMAR,6X,3HAPR,6X,3HMAY,6X,3HJUN,6X,3HJUL,6X,3HAUG,6X,3HSEP)
 3020 FORMAT(' ',3X,I2,12(1X,F8.4))
 4000 FORMAT(' ')
 4002 FORMAT(' ',12(1X,F8.0),1X,F11.0)
 4009 FORMAT(' ',12(1X,F8.0),1X,F11.0)
      YEARST=YEARST+1                                                   ANM186
      K=0                                                               ANM187
      GO TO 9100                                                        ANM188
  500 IF(NOPT.GE.1)GO TO 510                                            ANM189
      CALL ACDSTR                                                       ANM190
      GO TO 100                                                         ANM191
  510 IF(NOPT.GE.2)GO TO 8000                                           ANM192
      CALL ACDSTR                                                       ANM193
      GO TO 9100                                                        ANM194
 9200 CALL EXIT
      END
```

REFERENCES FOR APPENDIX B

1. Brown, W. E., "The Control of Acid Mine Drainage Using an Oxygen Diffusion Barrier," M.S. Thesis, The Ohio State University, (1970).

2. Chow, K. Y., "Computer Simulation of Acid Mine Drainage," M.S. Thesis, The Ohio State University, (1970).

3. General Assembly of the State of Ohio, Amended Substitute House Bill No. 928, "An Act," File No. 239, April 10, 1972.

4. Good, D. M., "The Relation of Refuse Pile Hydrology to Acid Production," M.S. Thesis, The Ohio State University, (1972).

5. Hill, R. D., Chief of Mine Drainage Pollution Control Activity, Federal Envinronmental Protection Agency, Personal Communication (1973).

6. Linsley, Jr., R. K., Kohler, M. A., Paulhus, J. L. H., Hydrology for Engineers, McGraw-Hill Book Company, Inc., New York (1958).

7. Mihok, E. A., Moebs, N. N., "U.S. Bureau of Mines Progress in Mine Water Research," Fourth Symposium on Coal Mine Drainage Research, Pittsburgh, (1972).

8. Morth, A. H., "Acid Mine Drainage: A Mathematical Model," Ph.D. Dissertation, The Ohio State University, (1971).

9. Ricca, V. T., Associate Professor, The Ohio State University, Personal Communication (1973).

10. Ricca, V. T., "The Ohio State University Version of the Stanford Streamflow Simulation Model," The Ohio State University, Water Resources Center (1972).

11. Ricca, V. T., Chow, K. Y., "Acid Mine Drainage Quantity and Quality Generation Model," The Ohio State University, (1973).

12. Rice, P. A., Rabolini, F., "Biological Treatment of Acid Mine Water," Fourth Symposium on Coal Mine Drainage Research, Pittsburgh, (1972).

13. Smith, E. E., Professor, The Ohio State University, Personal Communication (1973).

14. Smith, E. E., Shumate, K. W., "Development of a Natural Laboratory for the Study of Acid Mine Drainage Production," Second Symposium on Coal Mine Drainage Research, Pittsburgh, (1968).

15. Smith, E. E., Shumate, K. S., "Pilot Scale Study of Acid Mine Drainage," Program #14010 EPA Contract 14-12-97 (1971).

16. Sternberg, Y. M., Agnew, A. F., "Hydrology or Surface Mining -- A Case Study," Water Resources Research, (1968).

17. "Surface Mining and Our Environment," U.S. Department of Interior, (1967).

18. 91st Congress, S. 1075, Public Law 91-190, "National Environmental Policy Act of 1969," January 1, 1970.

19. Maupin, A. N., "Computer Simulation of Acid Mine Drainage from a Watershed Containing Refuse Piles and/or Surface Mines," M.S. Thesis, The Ohio State University, (1973).

20. Bird, R. S. et al. Transport Phenomena, New York (1960).

APPENDIX C

OPTIMIZATION MODEL FOR RESOURCE ALLOCATION TO ABATE MINE DRAINAGE POLLUTION

In preceding sections, models and procedures have been described for predicting mine drainage pollution levels, stream flow quantities, and costs to implement actions for reducing the effects of mine drainage pollution. Several distinct methods for reducing or eliminating the effects of mine drainage pollution have been outlined, and these methods are as diverse as sealing and flooding to slow pyrite oxidation and treatment to neutralize the acid in mine drainage effluent. For each method, both the effects on stream quality and the implementation costs will vary from site to site. Moreover, stream quality is a dynamic phenomenon in that weather will cause large variations in stream quality. Based upon inputs from the physical models of pollutant sources and the cost model, the optimization model has two principal objectives in analyzing the allocation of resources to control mine drainage pollution in a watershed:

1. Determine a least cost allocation to achieve a specified quality level, and

2. Determine the most effective allocation for a specified cost.

The initial research task in designing an optimization model to achieve the above objectives was directed toward the construction of a dynamic model of a single-stream basin having multiple sources of mine drainage pollution. The model is dynamic in that pollutant and stream flows are functions of time; however, these functions are regarded as deterministic. The effects and costs of resource allocation to reduce pollutant effects are represented in this dynamic single-stream model which will be identified by the mnemonics DSS. An analysis of DSS indicates the types of optimization models which will be required to represent the resource allocation options available with varying levels of realism. Two optimization models are developed and they are defined below along with their identifying mnemonics:

1. Deterministic "worst-case" minimum cost (DWMC) model,

2. Deterministic "worst-case" maximum effectiveness (DWME) model,

A detailed description of the DWMC and DWME models is contained in this appendix along with computer program flow charts and input data instructions. Because the DSS model served as a basis for the two optimization models mentioned above, the DSS model is described first.

DYNAMIC SINGLE-STREAM MODEL (DSS MODEL)

Consider a watershed having N mine drainage pollution sources, and resource allocation strategies for this watershed are to be determined to satisfy two different objectives, viz., 1) minimum cost to maintain a desired stream quality level and 2) maximum quality for a fixed cost budget. Initially a model to satisfy objective 1 above is outlined and then later this model is extended to satisfy objective 2. To illustrate the proposed optimization methods, a simplified stream is considered having no tributaries since the extension to include tributaries is straightforward. The watershed is illustrated in Figure C.1.

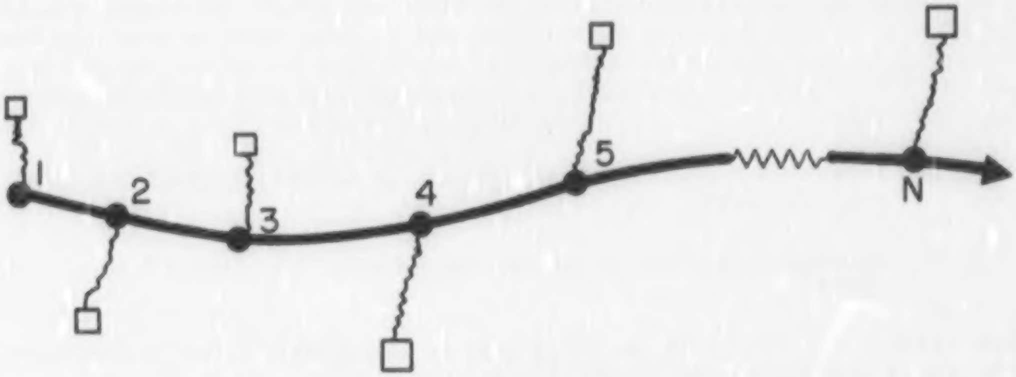


Figure C.1. Single-stream watershed

The first step in the model formulation process will be to define basic flows and decision variables. Then, these variables will be used to formulate criteria and constraint equations for the two objectives (minimum cost and maximum effectiveness).

Basic Flow and Decision Variables

Each mine, noted as a small square in the figure, is represented as draining into the stream at a node point, defining N node points. Without action to control pollutant effluent, the ith mine produces pollutants at a rate of $P_i(t)$ kilograms per hour at time t (hours) where $i = 1, 2, \ldots, N$. For example, the pollutant might be acid and $P_i(t)$ would specify the total acidity of the effluent from mine i. Negative values of $P_i(t)$ would specify an alkaline condition. Also the stream carries a fluid flow, exclusive of pollutant, of $Q_i(t)$ (kg per hour) at node point i at time t. The value of $Q_i(t)$ is assumed to be unaffected by actions to control stream pollution levels.

In addition to pollutant inputs from mine sources, the stream has natural pollutant inputs occurring throughout its length. These natural inputs are assumed to be distributed continuously between nodes, but the stream reach between each pair of nodes may have its own unique input rate. Let $P_i^n(t)$ be the natural pollutant input rate in kg per hour occurring between nodes i-1 and i. For example, a natural acid input rate of -.05 kg per hour between two nodes would indicate an alkaline condition alleviating part of any potential acid mine drainage.

A survey of possible methods for controlling mine drainage pollution indicated that these methods can be classified into three categories as far as the optimization model is concerned. These categories are:

1. abatement at the mine site,
2. treatment at the mine site, and
3. treatment in the stream channel.

Abatement is assumed to reduce the pollutant flow but not necessarily eliminate it. That is, site i produces a flow of $P_i^a(t)$ if abatement is performed at site i, where $P_i^a(t) < P_i(t)$ for all t. Examples of abatement are flooding or sealing of deep mines; covering, leveling, compacting, burying, or grading of gob piles; and grading, covering, or replanting of strip mines. All of these methods have the potential for reducing pollutant flows, but their effectiveness will vary from site to site.

Treatment, whether in the stream channel or at a site, is assumed to reduce pollutant flow to zero (without affecting the stream flow exclusive of pollutant; i.e., values of $Q_i(t)$), but treatment will not affect a condition where the stream pollutant measure is already negative. Using our acid mine drainage example again, the treatment facility will neutralize an acid stream until it has zero total acidity, but it will not affect an alkaline stream. The assumption is being made here that once the decision is made to install a treatment facility the most economical solution is to remove all acid conditions but do not change alkaline conditions. The effect of a treatment facility is shown graphically in Figure C.2. The performance of a treatment facility can be represented mathematically by a clip function, L(x),

$$\text{where } L(x) = \begin{cases} 0 \text{ if } x \geqslant 0 \\ x \text{ if } x < 0. \end{cases}$$

Thus, $L\left[P_i(t)\right]$ would represent the output of a mine source having a treatment facility.

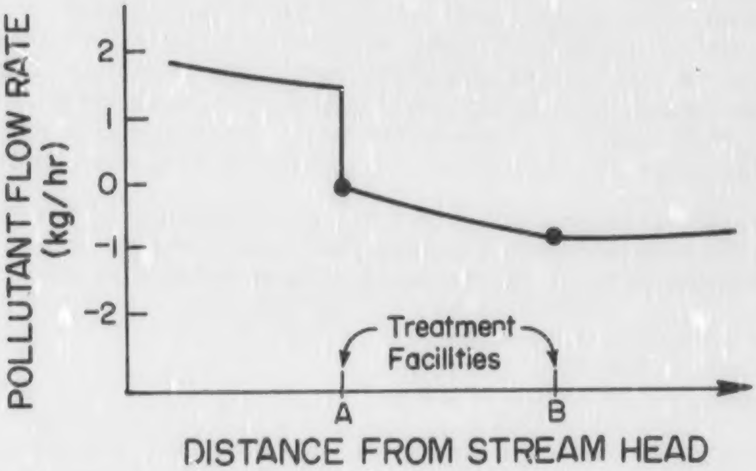


Figure C.2. Effect of instream treatment facilities

Three decision variables are used at each node to specify the pollution control measures to be used, if any. They are:

$$d_i^a = \begin{cases} 1 \text{ if abatement is done at site i} \\ 0 \text{ if otherwise} \end{cases}$$

$$d_i^t = \begin{cases} 1 \text{ if site treatment is done at site } i \\ 0 \text{ if otherwise} \end{cases}$$

$$d_i^s = \begin{cases} 1 \text{ if instream treatment is done at site } i \\ 0 \text{ if otherwise} \end{cases}$$

Site treatment, if implemented, is regarded as being capable of treating all pollutant effluent before it reaches the stream. Also, the assumption is made that instream treatment facilities treating all of the fluid passing through a node adjacent to a mine site could be located at any node; moreover, a mine site having an instream treatment facility would treat the effluent from its local mine source before its effluent enters the stream channel.

CONSTRAINT EQUATIONS AND CRITERION FUNCTION FOR MINIMUM COST MODEL

The minimum cost model selects values for the three decision variables, defined above, at each node in order to maintain stream quality throughout the watershed while minimizing total cost. The constraint equations guarantee that quality is maintained, and the criterion function specifies the minimum cost objective. The constraint equations are presented first.

The quality standard specifies that the pollutant concentration must be less than Q_s in parts per million (ppm) throughout the watershed. We will regard this to mean that the standard can be satisfied by having a quality of Q_s or better at each node. This assumption implies that satisfactory quality just downstream of node i together with a natural pollutant input between nodes i and i+1 will not violate the standard so long as quality is maintained just down stream of node i+1. If $P_i^t(t)$ is the total pollutant flow rate at node i (after considering all upstream sources, natural pollutant inputs, and abatement and treatment procedures), then

$$\frac{P_i^t(t) \cdot 10^6}{P_i^t(t) + Q_i(t)} \leq Q_s \quad \text{or}$$

$$P_i^t(t) \leq \frac{Q_s}{10^6 - Q_s} Q_i(t) \tag{C.1}$$

for $i = 1, 2, \ldots, N$ and for all values of t. Note that this quality standard must be met continuously throughout time.

The principal task in constructing the constraint equations is to specify $P_i^t(t)$. Consider the node i. Decisions with respect to the source are specified by the variables d_i^a and d_i^t. The mine pollution output rate after considering the abatement decision is given by

$$d_i^a P_i^a(t) + (1 - d_i^a) P_i(t) \,.$$

Including the site treatment decision, the pollution output rate from a mine source is defined as $P_i^s(t)$, and is given by

$$\begin{aligned} P_i^s(t) = {} & d_i^t L\left[d_i^a P_i^a(t) + \left(1 - d_i^a\right) P_i(t)\right] \\ & + \left[1 - d_i^t\right]\left[d_i^a P_i^a(t) + \left(1 - d_i^a\right) P_i(t)\right] \\ & \qquad i = 1, 2, \ldots, N \qquad \text{(C.2)} \end{aligned}$$

In addition to the site decisions, d_i^a and d_i^t, the pollutant flow from node i is a result of the interaction among the pollution flow rates, $P_i^t(t_i - \tau_{i-1})$ just downstream of node i-1, the natural pollutant input rate $P_i^n(t)$, and the instream processor decision d_i^s. τ_{i-1} is the time delay for a particle of water to flow from node i-1 to node i and is assumed to be constant for all values of t. The value of $P_i^t(t)$ is given by

$$\begin{aligned} P_i^t(t) = {} & d_i^s L\left[P_i^n(t) + P_{i-1}^t(t - \tau_{i-1}) + P_i^s(t)\right] \\ & + \left[1 - d_i^s\right]\left[P_i^n(t) + P_{i-1}^t(t - \tau_{i-1}) + P_i^s(t)\right] ; \\ & \qquad i = 1, 2, \ldots, N \qquad \text{(C.3)} \end{aligned}$$

where $P_0^t(t) = 0$.

Substitution of C.3 into C.1 gives the complete set of constraint equations. Notice that these constraint equations are nonlinear functions of the decision variables d_i^a, d_i^t, d_i^s.

In order to specify the minimum cost criterion function, several cost variables need to be defined. All costs incurred by these pollution control measures are assumed to be equivalent annual costs which include recovery of capital, operating and maintenance costs. If abatement is performed at site i, a cost of C_i^a is incurred. Annual treatment costs involve two cost components; i.e., a variable cost that is directly proportional to the annual amount of pollutant processed and the equivalent annual cost exclusive of the variable cost component. An example of the variable cost component would be the cost of chemicals for neutralization of acid. For treatment processors the following cost variables are used:

C_i^t = annual cost for a treatment processor at mine site i exclusive of the variable cost (dollars),

C_i^s = annual cost for an instream treatment processor at node i exclusive of the variable cost (dollars), and

C_v = variable cost to treat one unit of pollution (dollars per kg).

To determine the variable annual costs for treatment, the following additional annual pollutant loads are required as input:

A_i = annual pollutant load emitted from source i without site abatement (kg),

A_i^a = annual pollutant load emitted from source i with site abatement (kg)

A_i^n = annual natural pollutant input between nodes i-1 and i (kg).

Note that values for A_i^n and A_i^a may be negative indicating a flow of a substance that can "neutralize" the pollutant. The assumption is made that the effluent A_i^a from a mine site after abatement is either continuously positive or negative and does not alternate between positive and negative values. Similarly, the assumption is made that any negative pollutant inputs to the stream for all values of d_i^a, d_i^t, d_i^s will not cause the stream at a particular location to alternate between positive and negative states. This assumption is made so that the above annual pollutant inputs can be added at instream treatment processors to calculate annual variable treatment costs. Recall that only positive values of pollutant flow are treated at a treatment processor.

Using the costs and annual pollutant flows defined above, the total resource allocation cost, C_t, can be calculated by

$$C_t = \sum_{i=1}^{N} \Big[d_i^a C_i^a + d_i^t \Big[C_i^t + C_v \Big((1 - d_i^a) A_i - d_i^a L(-A_i^a) \Big) \Big] + d_i^s \Big[C_i^s - C_v L(-A_i^t) \Big] \Big] \qquad (C.4)$$

where A_i^t is the annual pollutant load passing stream node i considering the upstream decision variables d_j^a, d_j^t, d_j^s; for $j = 1, 2, \ldots, i$.

Values for A_i^t are calculated in a manner similar to $P_i^t(t)$. First, the annual inputs from individual mine sources are given by A_i^s and calculated by

$$A_i^s = d_i^t L\Big[d_i^a A_i^a + (1 - d_i^a) A_i \Big] + (1 - d_i^t)\Big[d_i^a A_i^a + (1 - d_i^a) A_i \Big] \qquad i = 1, 2, \ldots, N \qquad (C.5)$$

These values of A_i^s are inserted in the following expression for A_i^t.

$$A_i^t = d_i^s L\big(A_i^n + A_{i-1}^t + A_i^s \big) + (1 - d_i^s)\big(A_i^n + A_{i-1}^t + A_i^s \big) ; \qquad i = 1, 2, \ldots, N \qquad (C.6)$$

where $A_0^t = 0$.

Equations C.4 and C.1 with their inputs C.2, C.3, C.5, and C.6 complete the formulation of the minimum cost version of the DSS model. Equation C.4 is the criterion function, and the value of C_t given by that equation is to be minimized by choice of values for d_i^a, d_i^t, d_i^s, for $i = 1, 2, \ldots, N$. Of course, the constraint equations given by C.1 must be satisfied for all values of C_t considered.

CONSTRAINT EQUATIONS AND CRITERION FUNCTION FOR THE MAXIMUM EFFECTIVENESS MODEL

The purpose of the maximum effectiveness model is to allocate a fixed budget in the most effective manner. The effectiveness measure has been designed to indicate the relationships between pollution levels, environmental impact, and land use in the vicinity of the watershed. These relationships are reflected in the effectiveness measure using two concepts which are:

1. The basic value of a stream based upon maximum pollution concentration level.

2. The relative importance of the stream between two nodes based upon its land use.

The above concepts are applied to individual stream reaches between adjacent stream nodes to compute a reach effectiveness measure which is the product of its relative importance and its basic value. Total watershed effectiveness is assumed to be the sum of the individual reach effectiveness measures. These concepts are described in more detail below and illustrated by examples. Following the examples the mathematical notation is developed for the criterion function and constraint equations.

The basic value of a stream reach is a number between zero and ten that indicates the reach's value based on observable effects from pollution concentrations. These effects include phenomena such as aquatic life, aesthetics, and water supply processing which are assumed to be related to the maximum pollution concentration experienced. To portray the variation in basic value, maximum pollution concentrations are classified into intervals within which the observable effects are assumed to be constant. Table C.1 illustrates the variation of basic value with pollution concentration.

For each stream reach between two nodes, the basic value is determined and is weighted by the relative importance of the stream reach to give the effectiveness measure for this same stream reach. Relative importance is a quantity varying between zero and ten that specifies the importance of controlling pollution levels in each stream reach between adjacent nodes, land use in the vicinity of the stream reach, and the impact of pollution on this land use. Also, the length of the reach can be considered in assigning relative importance values. The distribution of relative importance values throughout the watershed is determined in several steps. The first step is to select the most important stream reach (as defined by the stream between two adjacent nodes). The most important stream reach is given a relative importance value of ten. Then all other stream reaches are assigned values consistent with the difference between their importance and the importance of the most important stream reach.

The application of the above concepts is illustrated by the stream portrayed in Figure C.3. The predominate land uses are noted in the figure. The most important reach with respect to the impact of pollution is the reach between nodes 1 and 2; thus, this reach is assigned a relative importance of 10.0. The other reaches are evaluated to have relative importances of 7.0 downstream of node 4, 5.0 between nodes 2 and 3, and 2.0 between nodes 3 and 4. Using these relative importances, the effectiveness measures for each stream reach can be obtained by

Table C.1. ILLUSTRATIVE VARIATION OF BASIC VALUE WITH MAXIMUM POLLUTION CONCENTRATION (ppm)

Maximum pollution concentration	Observable effects	Basic value
> 10	Fish cannot survive, noticeable odor, strong discoloration, water treatment costs increased by 100%.	0
8-10	Game fish cannot survive, high scavenger fish mortality, noticeable odor, water treatment costs increased by 50%.	2
6-8	High game fish mortality, scavenger fish will not reproduce, water treatment costs increased by 25%.	4
4-6	Game fish will not reproduce.	7.5
< 4	Aquatic life unaffected.	10

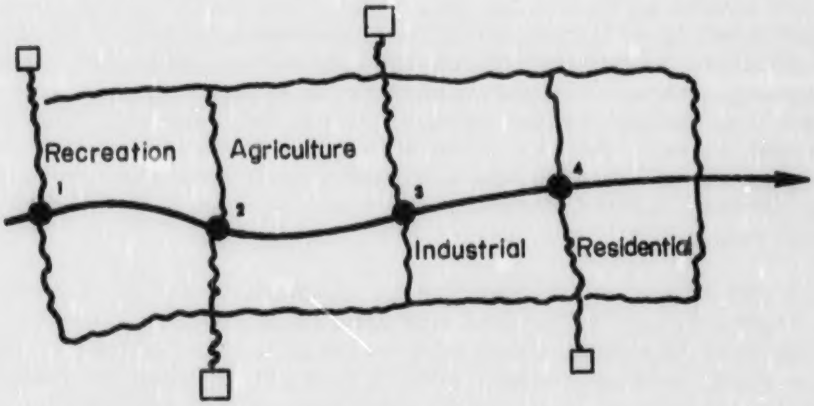


Figure C.3. Single stream with adjoining land uses

determining basic values for each reach and multiplying these basic values by their respective relative importances. For specified values of d_i^a, d_i^t, and d_i^s, let the maximum pollution concentrations be 6.8, 5.9, 5.5, 6.1 ppm starting at the head of the stream (node 1) and proceding downstream. Then using Table C.1, the basic values for each reach are 4, 7.5, 7.5, and 4, respectively. Multiplying by the reach relative importances, the individual reach effectiveness measures are 40.0, 37.5, 15.0, and 28.0, respectively. Summing these reach effectiveness measures, the stream effectiveness measure is 120.5.

Having presented an example, the mathematical notation necessary to use the effectiveness measure will now be defined.

Let E^j = the basic value for the jth maximum pollutant concentration interval, $0 \leq E^j \leq 10$; $j = 1, 2, \ldots, N^*$;

N^* = total number of pollutant concentration intervals;

Q^j = the upper limit on the maximum pollutant concentration for the jth interval; $j = 1, 2, \ldots, N^*$;

$Q^0 = 0$;

R_i = relative importance of the stream reach between nodes i and i+1, $i = 1, 2, \ldots, n$; $0 \leq R_i \leq 10$;

E_i = effectiveness of pollution control actions on the stream reach between nodes i and i+1, $i = 1, 2, \ldots, N$.

To compute the basic value of reach i, the maximum pollutant concentration over all time values is determined, and this value is used to determine the pollutant concentration interval. That is, the value of j is determined for reach i such that

$$Q^{j-1} < \max_t \frac{P_i^t(t) \cdot 10^6}{P_i^t(t) + Q_i(t)} \leq Q^j \tag{C.7}$$

using this value of j then

$$E_i = R_i \cdot E^j . \tag{C.8}$$

After determining values of E_i for $i = 1, 2, \ldots, N$; then, the total watershed effectiveness, E, is computed by

$$E = \sum_{i=1}^{N} E_i \qquad (C.9)$$

The above equation is the criterion function for the maximum effectiveness model.

Since the constraint for the maximum effectiveness model is the total annual cost budget, B, then the total resource allocation cost, C_t, given by equation C.4 can be used in the constraint equation. It follows that

$$C_t \leq B \qquad (C.10)$$

ANALYSIS OF THE DYNAMIC SINGLE-STREAM MODEL

Analysis of the minimum cost and maximum effectiveness models formulated in this section shows that both optimization problems have the following characteristics:

1. set of admissable decisions is discrete,
2. number of possible decisions is 8^N,
3. constraint equations and criteria functions are nonlinear functions of the decision variables.

Note that the number of possible decisions will be overwhelming with values of N as large as 20 or 30. Even with only 10 mine sources the the number of possible decisions is in excess of one billon.

In addition to the characteristics mentioned above, another factor complicating any solution procedure is the dynamic stochastic nature of pollutant and stream flows. That is, the functions $P_i(t)$, $P_i^n(t)$, $P_i^a(t)$, and $Q_i(t)$ for $i = 1, 2, \ldots, N$ are time varying and stochastic. These functions will be difficult to handle in equations such as C.1 and C.2 even if they are regarded as deterministic. Since these functions are strongly influenced by precipitation patterns, they are in fact stochastic. One way of handling their stochastic nature is to adopt a conservative approach by selecting very long time traces from the Pollutant Source and Hydrologic Models; or, even better, analyses of the pollutant source models can be conducted to indicate which precipitation patterns give the maximum pollution concentrations. In this way, confidence can be gained in the fact that quality constraints will be maintained in the minimum cost model (equation C.1) and that effectiveness measures will be properly calculated (equation C.7). The basic assumption being made is that optimizing in this manner will yield a solution that provides quality and/or effectiveness measures at least as good as the worst case analyzed so frequently that any violations can be ignored.

Going one step further, a much less cumbersome and more useful optimization algorithm can be obtained by completely suppressing the dynamic stochastic nature of the functions $P_i(t)$, ${P_i}^n(t)$, ${P_i}^a(t)$, and $Q_i(t)$. This approach will be called the "deterministic worst-case analysis." The basic idea inherent in this approach is to select a set of single values from the functions $P_i(t)$, ${P_i}^n(t)$, ${P_i}^a(t)$, $Q_i(t)$, $i = 1, 2, \ldots, N$; that represents the most adverse situation from a quality viewpoint. Then, an optimal solution using these values should almost always give better quality or more effectiveness in actual practise than considered in the solution procedure.

Two models have been developed to evaluate the deterministic worst-case approach. These models are the DWMC and DWME models defined in the introduction to this appendix. Both models can analyze a basin having multiple streams with tributaries, and the method for representing mine sources and streams in this basin is described in the following section. The deterministic models are described in detail in subsequent sections.

BASIN MINE SOURCE AND STREAM DESIGNATION SYSTEM

In this section, notation used to define the basin stream network, mine sources, pollutant flows, and pollution control costs is defined. This notation is used in both of the deterministic worst case optimization programs, and standard FORTRAN symbols are used to designate variables to avoid use of an additional set of mathematical notations.

The basin is assumed to have mine sources draining into a stream network having a hierarchy of at most level three. That is, the basin outlet is a third-level stream being fed by second-level streams, and the second-level streams are fed by first-level streams. Note that only second-level streams can feed the third-level stream and that only one third-level stream is permitted. Of course, basins with only a single stream or with only a hierarchy of level two can be represented by the optimization models. In these cases the basin outlet stream would still be a level three stream and there would be no level one streams. In the single stream case there would only be a single level three stream.

A set of nodes are denoted along each stream and there exists three different functions that can be performed at each node; i.e., a pollution source, an instream treatment site, or a stream confluence. All nodes are source nodes or possible points at which pollutants may enter the stream from mines. Any pollutant flow from mines is assumed to enter the stream at the node location. In addition natural pollutant inputs may occur between stream nodes. These natural pollutant inputs may be positive, increasing pollutant concentrations, or negative, neutralizing pollutant concentrations. Quality checks are made at each node to determine whether pollutant concentrations are less than the standard or to determine the effectiveness of a given resource allocation. These quality checks are always made just downstream of each node.

Treatment nodes are possible locations for instream treatment facilities removing pollutants from the entire stream flow. The DSS model permitted all nodes to have the potential for an instream treatment facility; however, economies with respect to the number of alternatives to evaluate are achieved by evaluating only the most likely locations. It should be noted that the models can be operated with all nodes being treatment nodes if the additional computer time is considered worthwhile. A mine source at a treatment node is assumed to feed the instream treatment facility directly, if the decision is made to implement the facility; thus, the source at an instream treatment site would not degrade stream quality.

In addition, some nodes, called confluence nodes, are fed by another lower level stream. An example is shown in Figure C.4 where nodes 4 on second level stream number 1, 2 on the third level stream, and 4 on the third level stream are confluence nodes. Note that the confluence is located between the node and the next upstream node. Also, nodes 5 on second level stream number 1 and 3 on the third level stream are treatment nodes.

Each stream is designated by its level and streams having the same level are numbered. Thus, for level 1 streams, the streams are numbered one through NS(1) where NS(1) is the total number of level one streams. Similarly, level two streams are numbered one through NS(2). There is only one level 3 stream; i.e., NS(3) = 1. Also, the variable KL is used to designate the lowest level stream represented in the basin. If the basin has only two levels in its stream network, then NS(1) = 0 and KL = 2. Moreover, a single stream network would have NS(1) = NS(2) = 0 and KL = 3.

As depicted in Figure C.4, the nodes on each stream are numbered starting with the most upstream node. Thus, node I on stream J is upstream of I+1 on stream J. Using the three coordinates; i.e., node number I, stream number J, and stream level K; an individual node is uniquely specified within the basin by the ordered triple (I,J,K). The array

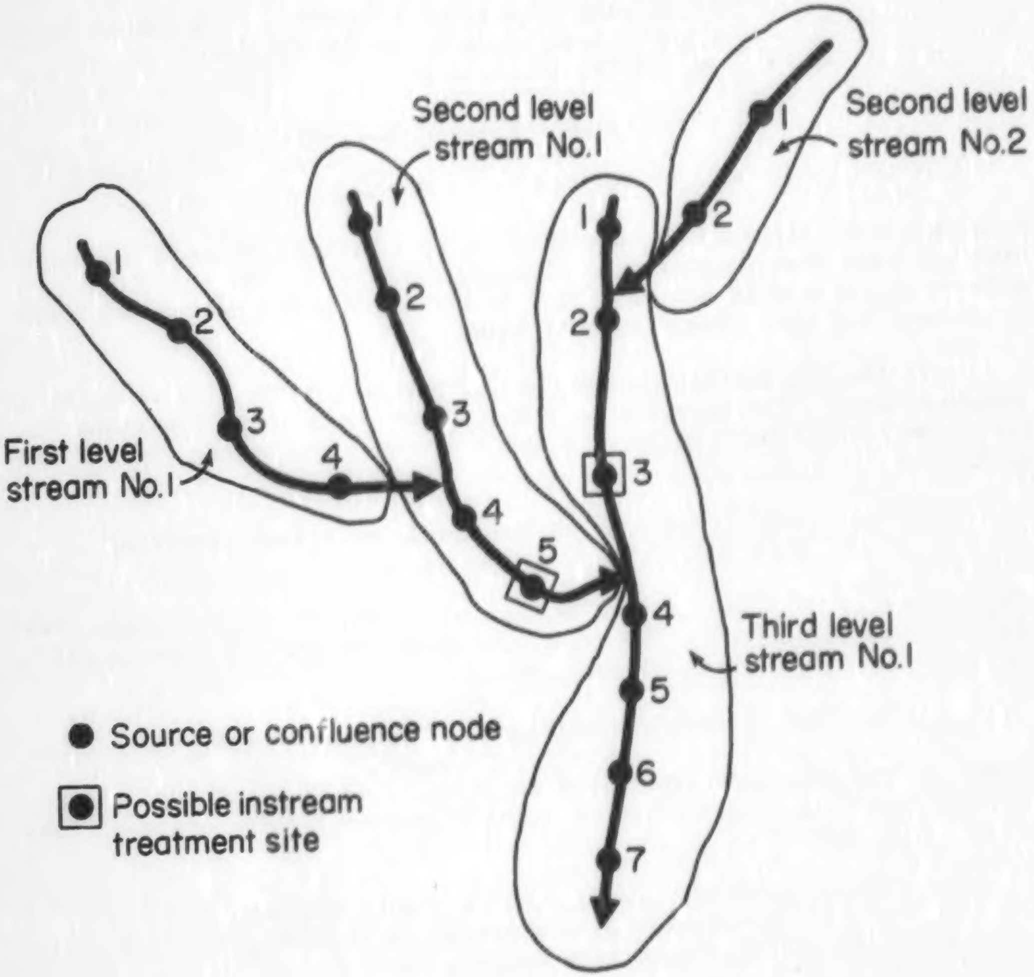

Second level stream No.1
Second level stream No.2
First level stream No.1
Third level stream No.1
1
2
3
4
5
6
7
● Source or confluence node
Possible instream treatment site

Figure C.4. Illustration of stream and node designation

ND specifies the number of nodes on each stream. Values for ND are specified as input data and are defined by

ND(J,K) = number of nodes on stream J of level K.

Several arrays are employed to permit tracing from a level one stream to the basin outlet, or backwards from the basin outlet to the stream heads. The array JN permits tracing upstream and is prepared as input data to the optimization models. Values for JN are defined by

$$JN(I,J,K-1) = \begin{cases} \text{NF if node I on level K stream J is a confluence node where NF is the level K-1 stream feeding J, and} \\ \text{0 if (I,J,K) is not a confluence node} \end{cases}$$

for K = 2,3.

Also JN(I,J,0) will be defined to be zero.

Note that the last node or node ND(NF,K-1) on stream NF would be the next upstream node feeding node (I,J,K). Also note that the first node on any stream is not permitted to be a confluence node since there is no need for two streams in this case.

To permit tracing downstream through the stream network, the optimization programs create two arrays from the array JN. The array NFN gives the nodes receiving flow from level 1 and 2 streams, and values for NFN are defined by

NFN(J,K) = confluence node on the level K+1 stream receiving flow from level K stream J.

Since there is only one level 3 stream, all level two streams feed level three stream number one. The array NFS gives the level 2 streams receiving flow from level 1 streams, where

NFS(J) = level 2 stream receiving flow from level 1 stream J.

Potential instream treatment sites are designated in a manner similar to that used to denote confluence nodes. Treatment nodes are specified by the array INT, where

$$INT(I,J,K) = \begin{cases} \text{NT if node (I,J,K) is a potential instream treatment site where NT is the instream treatment site number, and} \\ \text{0 if otherwise} \end{cases}$$

NT can take on the values between one and NINST where NINST is the total number of instream treatment sites considered.

At each node a set of variables is used to depict the stream flow, pollutant flows, and costs associated with each pollution control measure. Recall that the deterministic worst case optimization models are based on the assumption that pollutant and stream flows are selected to represent the most adverse situation from a pollution viewpoint. It is assumed that the worst case has been identified for the variables defined below. The stream flow at each node is given by

Q(I,J,K) = stream flow at node (I,J,K) exclusive of pollutant.

Input values for Q(I,J,K) are in cubic meters per second, and the program converts these amounts to kilograms per hour for the computational procedure. The pollutant flow rates for each mine source are given as input by

P(I,J,K) = pollutant flow emitted from the source at node (I,J,K) without implementation of any pollution control measures (kilograms per hour), and

PA(I,J,K) = pollutant flow emitted from the source at node (I,J,K) if abatement is implemented (kilograms per hour).

Also, the natural pollutant inputs are specified as input data by

PN(I,J,K) = pollutant input to the stream between nodes (I-1,J,K) and (I,J,K) due to natural sources (kilograms per hour).

If I=1, PN(I,J,K) represents all of the natural pollutant input above the first node. If (I,J,K) is a confluence node, PN(I,J,K) is the natural pollutant input between (I-1,J,K) and (I,J,K) plus the natural pollutant input between stream (J,K) and the last node on stream JN(I,J,K-1).

Corresponding to these "instantaneous" pollutant flow rates given above, a set of mean annual pollutant flows are required to calculate treatment variable costs. These annual pollutant flows are specified as input data by

AP(I,J,K) = mean annual pollutant load emitted by the source at node (I,J,K) if no pollution controls are implemented (kilograms)

APA(I,J,K) = mean annual pollutant load emitted by the source at node (I,J,K) if abatement is implemented (kilograms)

APN(I,J,K) = mean annual pollutant input due to natural sources between node (I,J,K) and the next upstream node (or nodes when (I,J,K) is a confluence node) in kilograms.

In addition to pollution and stream flows, a set of costs for each alternative control measure at each node is specified as input. The array C gives the equivalent annual costs for abatement and treatment (exclusive of variable chemical costs), and values for the array C are defined by

$$C(ID,I,J,K) = \begin{cases} \text{annual costs to abate the mine source at } (I,J,K) \text{ for } ID = 1\ (\$) \\ \text{annual cost to implement a treatment processor for the mine source at } (I,J,K) \text{ for } ID = 2(\$) \end{cases}$$

The equivalent annual costs to implement instream treatment processors are specified by the array CI, where

CI(NT) = equivalent annual fixed cost to implement instream treatment at instream treatment site number NT ($)

for

NT = 1, 2, ..., NINST; and

CI(0) = 0.

The variable chemical cost for treatment processors is given by VC, in dollars per kilogram, specified as input data.

DETERMINISTIC "WORST-CASE" MAXIMUM EFFECTIVENESS (DWME) MODEL

In this section, the DWME model is described, and the algorithm for determining the maximum effectiveness solution is presented. This algorithm is performed by program MAXEF to determine a maximum effectiveness solution. To simplify the understanding of MAXEF, the same notation is used to describe the algorithm as is used in the program. Thus, FORTRAN variable names are employed in this section. Computer program flow charts and input data instructions are contained in a later section of this appendix.

Criterion Function and Constraint Equation

The DWME model is specified by its criterion function and constraint equation. These equations are obtained simply by converting the corresponding equations for the DSS model to represent a network of streams and to incorporate the "worst-case" pollutant and stream flows defined in the previous section.

The DWME analogue of the criterion function given by C.9 is

$$EFF = \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} \sum_{I=1}^{ND(J,K)} E(I,J,K)\,, \qquad (C.11)$$

where E(I,J,K) = effectiveness of pollution control actions on the stream reach between node (I,J,K) and the next downstream node on stream J.

EFF = total basin effectiveness of a set of pollution control actions.

If (I,J,K) is the last node on stream J; i.e., I = ND(J,K), then the next downstream node is interpreted to be the outlet of stream J. Individual reach effectiveness values are determined by

$$E(I,J,K) = R(I,J,K) \cdot EJ(I,J,K) \qquad K = 1,2,3;\ J = 1,\ldots,NS(K);\ I = 1,\ldots,ND(J,K) \tag{C.12}$$

where R(I,J,K) = relative importance of the stream reach between node (I,J,K) and the next downstream node on stream J, $0 \leq R(I,J,K) \leq 10$;

EJ(I,J,K) = basic value of the stream reach between node (I,J,K) and the next downstream node on stream J. $0 \leq EJ(I,J,K) \leq 10$.

To compute the basic value of a stream reach, the pollution concentration as a result of a set of control actions is determined and compared with a set of pollution concentration intervals and their corresponding basic values, specified as input data. That is, M is determined so that it satisfies

$$QJ(M-1) < \frac{PLT(I,J,K)}{PLT(I,J,K) + Q(I,J,K)} \leq QJ(M)\ , \tag{C.13}$$

where QJ(M) = upper limit on the pollution concentration for the Mth interval (input values are in parts per million (ppm) but QJ(M) values are converted later to decimal fractions),

QJ(0) = 0

NI = total number of pollution concentration intervals

M = 1, 2, ..., NI, and

PLT(I,J,K) = pollutant flow rate in kilograms per hour just downstream at node (I,J,K) resulting from a specified set of control actions.

This value of M is used to specify the individual reach basic values by

$$EJ(I,J,K) = BV(M), \tag{C.14}$$

where BV(M) = basic value for pollution concentration interval M.

In order to determine the pollution flow from a set of control actions, values of PLT(I,J,K) must be determined. PLT(I,J,K) is the deterministic "worst-case" multi-stream analogue of $P_i{}^t(t)$ given by equation C.3, or the total pollutant flow just downstream of node (I,J,K). Pollutant may reach each node from one or more of the following sources:

1. flow past next upstream node,
2. natural flow originating downstream of next upstream node,
3. tributary if (I,J,K) is a confluence node, and
4. mine source at (I,J,K).

The flow past the next upstream node is given by PLT(I-1,J,K), and the case where I,J,K is at the head of a stream is handled by defining

$$PLT(0,J,K) = 0$$

for J = 1,2,...,NS(K);

K = 1,2,3.

The natural flow is given by PN(I,J,K). Pollutant flow from a tributary is represented by

PT(J,K) = pollutant output rate of stream J of level K (kilograms per hour);

where PT(J,K) = PLT(ND(J,K),J,K) if J > 0 and K > 0,

PT(0,K) = 0;

PT(J,0) = 0;
PT(0,0) = 0;
J = 1,2,...,NS(K); and

K = 1,2,3.

Also, pollutant output from a source is defined by

PS(I,J,K) = pollutant output rate from source (I,J,K) given the site abatement and treatment decisions (kilograms per hour).

In addition, pollutant may be removed from the stream by implementing an instream processor. Decisions to do so are specified by

$$\mathrm{DINT(NT)} = \begin{cases} 0 \text{ if instream processor site NT is implemented} \\ 1 \text{ if otherwise.} \end{cases}$$

$$\mathrm{NT} = 1,2,\ldots,\mathrm{NINST}.$$

If the node (I,J,K) cannot have an instream processor, then NT = INT(I,J,K) = 0 and DINT(0) = 1. Then necessity for defining DINT(NT) = 0 if the instream processor is implemented will be apparent when the optimization procedure is presented. Using the above relationships, the pollutant flow just downstream of node (I,J,K) is given by

$$\begin{aligned} \mathrm{PLT(I,J,K)} = {} & \mathrm{DINT[INT(I,J,K)]}\cdot[\mathrm{PN(I,J,K)} + \mathrm{PLT(I\text{-}1,J,K)} + \mathrm{PS(I,J,K)} \\ & + \mathrm{PT(JN(I,J,K\text{-}1),K\text{-}1)}] + (1 - \mathrm{DINT[INT(I,J,K)]}) \\ & \cdot \mathrm{L}[\mathrm{PN(I,J,K)} + \mathrm{PLT(I\text{-}1,J,K)} + \mathrm{PS(I,J,K)} \\ & + \mathrm{PT(JN(I,J,K\text{-}1),K\text{-}1)}] \end{aligned} \qquad (C.15)$$

for I = 1,2,...,ND(J,K);

J = 1,2,...,NS(K); and

K = 1,2,3.

Recall that L(X) is the clip function defined earlier.

The four equations above require values for the pollution output rate from a mine source after allowing for the site abatement or treatment decisions. Let these decisions be represented by

$$\mathrm{DA(I,J,K)} = \begin{cases} 1 \text{ if abatement is performed at mine source (I,J,K), and} \\ 0 \text{ if otherwise} \end{cases}$$

$$DT(I,J,K) = \begin{cases} 1 \text{ if site treatment is to be performed at site } (I,J,K), \text{ and} \\ 0 \text{ if otherwise.} \end{cases}$$

The deterministic "worst-case" multi-stream analogue of equation C.2 is used to compute values for PS(I,J,K). Thus,

$$PS(I,J,K) = DT(I,J,K) \cdot L(DA(I,J,K) \cdot PA(I,J,K) + [1-DA(I,J,K)] \cdot P(I,J,K)) + [1-DT(I,J,K)] \cdot (DA(I,J,K) \cdot PA(I,J,K) + [1-DA(I,J,K)] \cdot P(I,J,K)) \tag{C.16}$$

Equations C.11 through C.16 can be used to compute values for the criterion function of the DWME model. The cost constraint is

$$TTC \leq BUD , \tag{C.17}$$

where BUD = maximum allowable annual pollution control cost ($), and

TTC = annual pollution control cost expected as a result of decisions represented by DINT, DA, and DT arrays.

The total pollution control cost is calculated using the multi-stream "worst-case" analogue of equation C.4. That is,

$$TTC = \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} \sum_{I=1}^{ND(J,K)} (CS(I,J,K) + [1-DINT(INT(I,J,K))] \cdot [CI(INT(I,J,K)) - VC \cdot L[-APLT(I,J,K)]]) \; ; \tag{C.18}$$

where CS(I,J,K) = annual cost of pollution control decisions at mine source (I,J,K) in dollars;

APLT(I,J,K) = annual pollutant load in kilograms passing stream node (I,J,K) considering the upstream decision variables given by the DINT, DA, and DT arrays; and

APLT(0,J,K) = 0 .

Values of CS(I,J,K) are given by

$$\begin{aligned}CS(I,J,K) = {} & DA(I,J,K)\cdot C(1,I,J,K) + DT(I,J,K)\cdot[C(2,I,J,K) \\ & + VC\cdot([1-DA(I,J,K)]\cdot AP(I,J,K) \\ & - DA(I,J,K)\cdot L[\cdot APA(I,J,K)])]\end{aligned} \qquad (C.19)$$

Values of APLT(I,J,K) are determined in a manner quite similar to PLT(I,J,K). That is,

$$\begin{aligned}APLT(I,J,K) = {} & DINT[INT(I,J,K)]\cdot(APN(I,J,K) + APLT(I-1,J,K) + APS(I,J,K) \\ & + APT[JN(I,J,K-1),K-1]) + [1-DINT(INT(I,K,K))] \\ & \cdot L(APN(I,J,K) + APLT(I-1,J,K) + APS(I,J,K) \\ & + APT[JN(I,J,K-1),K-1]) ;\end{aligned} \qquad (C.20)$$

where APT(J,K) = annual pollutant output of stream J of level K in kilograms, [APT(J,K) = APLT(ND(J,K),J,K) if $J > 0$ and $K > 0$];

APT(0,K) = 0;

APT(J,0) = 0;

APS(I,J,K) = annual pollutant output rate from source (I,J,K) given the site abatement and treatment decisions (kilograms);

I = 1,2,...,ND(J,K);

J = 1,2,...,NS(K); and

K = 1,2,3.

The values for the annual source outputs are given by

$$\begin{aligned}APS(I,J,K) = {} & DT(I,J,K)\cdot L(DA(I,J,K)\cdot APA(I,J,K) + (1-DA(I,J,K)) \\ & \cdot AP(I,J,K)) + (1-DT(I,J,K))\cdot(DA(I,J,K)\cdot APA(I,J,K) \\ & + (1-DA(I,J,K))\cdot AP(I,J,K))\end{aligned} \qquad (C.21)$$

for I = 1,2,...,ND(J,K);

J = 1,2,...,NS(K); and K = 1,2,3.

Equations C.18, C.19, C.20, and C.21 can be used to evaluate the DWME cost constraint.

Optimization Algorithm

As pointed out by the DSS model, the criterion function and constraint equations for the DWME model are nonlinear functions of discrete variables, viz., the DINT, DA, and DT arrays. Thus, available optimization algorithms such as linear programming, nonlinear programming, game theory, decision theory, geometric programming, and stochastic programming are inappropriate. Dynamic programming is a possible solution method, but preliminary investigations indicated that computer requirements for a dynamic programming would be more extensive than the method selected. This conclusion results from the requirement to determine the optimal resource allocation function at each node for all admissible cost values. The algorithms developed as a result of this research (for the DWME and DWMC models) are modifications and extensions to the partial enumeration method for discrete optimization problems formulated by Lawler and Bell.[1]

The Lawler-Bell algorithm is designed to solve discrete optimization problems of the following form:

$$\begin{aligned} \text{Minimize} \quad & g_0(X), \\ \text{Subject to} \quad & g_{11}(X) - g_{12}(X) \geq 0, \\ & g_{21}(X) - g_{22}(X) \geq 0, \\ & \quad \vdots \qquad\quad \vdots \\ & g_{m1}(X) - g_{m2}(X) \geq 0, \end{aligned}$$

where $X = (x_1, x_2, \ldots, x_n)$, and

$x_j = 0$ or 1; $j = 1,2,\ldots,n$; and

where the restriction is applied that each of the functions $g_0(X)$, $g_{11}(X), \ldots, g_{m2}(X)$ is monotone nondecreasing in each of the variables $x_1, x_2, \ldots, x_n$.

The basic concept behind the Lawler-Bell algorithm involves ordering each possible vector X and then proceeding through the list of vectors to evaluate each vector which could be optimal. Naturally, those vectors which could not be optimal are skipped. A lexicographic or numerical ordering is obtained by giving each vector the integer value

---

[1] E. L. Lawler and M. D. Bell, "A Method for Solving Discrete Optimization Problems," Operations Research, Vol. 14, No. 6, Nov.-Dec. 1966, pp. 1098-1112.

$$n(X) = x_1 2^{n-1} + x_2 2^{n-2} + \cdots + x_n 2^0 .$$

In addition to the numerical ordering, a vector partial ordering is obtained by regarding

$$X \leq Y \text{ if and only if } x_j \leq y_j \text{ for } j = 1,2,\ldots,n$$

where $Y = (y_i, y_1, \ldots, y_n)$.

Note that $X \leq Y$ and $Y \nleq X$ are not equivalent expressions. Let X^* denote the first vector following X in the numerical ordering where

$$X \nleq X^* .$$

Lawler and Bell show that X^* can be obtained from X readily on a digital computer by

1. regarding X as a binary number,
2. subtracting 1 from X,
3. logically "or" X and $X-1$ to obtain X^*-1, and
4. adding 1 to obtain X^*.

The procedure described by Lawler and Bell to identify the optimal solution involves proceeding through the list of possible solutions and keeping a record of the least costly solution currently identified. $\hat{X}$ denotes this solution, and $g_0(\hat{X})$ is the minimum criterion function value. Procedure starts with $X = (0,0,\ldots,0)$ and ends when $X = (1,1,1,\ldots,1)$. Letting X denote the vector that is currently being examined, the following rules indicate which vector is evaluated after X:

1. If $g_0(X) \geq g_0(\hat{X})$, skip to X^*. Since X^* is the first vector in numerical order following X where $X \nleq X^*$, then $X+1, X+2, \ldots, X^*-1$ are all greater than or equal to X in the vector partial ordering. Also, because $g_0(X)$ is monotone nondecreasing, none of the vectors $X+1, X+2, \ldots, X^*-1$ can have values of $g_0(X)$ less than $g_0(\hat{X})$.

2. If X is a feasible solution satisfying all constraints and $g_0(X) < g_0(\hat{X})$, then X is substituted for $\hat{X}$.

3. If X is a feasible solution satisfying all constraints, skip to X^*. Because $g_0(X)$ is monotone nondecreasing, none of the vectors X+1, X+2, ..., X^*-1 can have values of the criterion function less than $g_0(X)$.

4. If $g_{i1}(X^*-1) - g_{i2}(X) \not\geq 0$ for any i=1,2,...,m, skip to X^*. Note that $Y = X^*-1$ maximizes $g_{i1}(Y)$ and $Y = X$ maximizes $-g_{i2}(Y)$ for any vector between X and X^*-1. Thus, if $g_{i1}(X^*-1) - g_{i2}(X) \not\geq 0$, then no vector Y between X and X^*-1 can satisfy constraint i.

5. Skip to X+1 if conditions 1, 3, and 4 do not apply.

Lawler and Bell give representative compute times to solve typical problems to illustrate the potential efficiency of their method. Using the procedure listed above, they solved problems involving as many as 30 binary variables with compute times ranging from 10 to 20 minutes on an IBM 7090. However, third generation computers are considerably faster than an IBM 7090. Although the increase in compute time for increasing numbers of variables appears to be less than an exponential function, it increases at a faster rate than a linear function. Thus, there appears to be an upper limit on the number of nodes than can be analyzed in a single problem, and it is important to select an algorithm that is rapid.

With emphasis upon this efficiency objective, the following extensions were accomplished in constructing the DWME optimization algorithm:

1. The number of possible alternatives at each mine source to be evaluated in the optimization algorithm was reduced from four to three. This modification saves considerable computer time because 4^n is much larger than 3^n where n is the number of possible mine sources.

2. The criterion function and constraint equations were modified so that the criterion function is equivalent to a nondecreasing function of each of the decision variables that is to be minimized.

3. The decision variables for a stream network were transformed so that they could be expressed as a vector X.

4. A procedure was developed for determining the next feasible solution subsequent to any point in the numerical ordering of the decision vector X. Note that condition 4

in the Lawler-Bell procedure defined above only allows for a test to be made to determine if decision vector values can be skipped and does not identify exactly the next feasible solution.

5. The last decision vector, noted as X_L, that needs to be evaluated in the numerical ordering to find the optimal solution was identified to reduce the number of iterations required. That is, decision vectors beyond X_L in the numerical ordering will not be optimal.

These extensions listed above are described next in the following discussion of the optimization algorithm.

Reduction of the Number of Alternatives at Each Mine Source -

The objective of reducing the total number of possible alternatives is to decrease the number of iterations by the optimization program to identify an optimal pollution. To illustrate the potential for improvement, consider a basin having three sources and no potential instream processing sites. Four alternatives exist at each source when one considers all possible combinations of site treatment and site abatement. Letting the variable MS denote these alternatives

$$MS = \begin{cases} 0 \text{ if no pollution control actions are to be performed} \\ 1 \text{ if abatement but no source treatment is to be performed} \\ 2 \text{ if treatment but no abatement is to be performed} \\ 3 \text{ if both treatment and abatement are to be performed} \end{cases}$$

Thus, direct enumeration of all possible combinations of control actions gives 4^3 or 64 alternatives to be considered. However, the pollutant output of a mine source, say (I,J,K), can have, at most, three possible values which are P(I,J,K), PA(I,J,K), and zero kilograms per hour if site treatment is performed. Thus, from a source output viewpoint there are only 3^3 or 27 alternatives to be considered as opposed to 64 originally.

Zero pollutant output can be obtained in two ways; i.e., site treatment alone and site treatment combined with abatement; and there are no interactions to be considered in evaluating these two ways with regard to stream quality. Obviously, the preferred alternative would be site treatment or site treatment combined with abatement depending on which is cheapest. Thus, the zero pollutant output alternative would be both site treatment and abatement if the annual cost of this alternative is less than site treatment alone. That is, if

$$C(1,I,J,K) + C(2,I,J,K) - VC \cdot L(-APA(I,J,K)) < C(2,I,J,K) + VC \cdot AP(I,J,K)$$

select site treatment and abatement for the zero pollutant output alternative. Otherwise, select site treatment alone.

From a mine output viewpoint, there are further potential reductions in alternatives to be considered. Some mines may cost more to abate than to treat. This may occur when sealing or other abatement procedures are potentially very expensive. Thus ignore the alternative of abatement alone if

$$C(1,I,J,K) \geq C(2,I,J,K) + VC \cdot AP(I,J,K) .$$

The result of the tradeoffs described above to reduce the number of basin alternatives is recorded in several arrays for ready reference during the optimization procedure. In addition to no pollution control, two control actions are considered and they are identified by the decision variable ID, where

$$ID = \begin{cases} 0 \text{ if no pollution control is to be implemented at a source} \\ 1 \text{ to identify the lowest cost pollution control alternative (usually involving abatement alone), and} \\ 2 \text{ to identify the pollution control alternative involving treatment} \end{cases}$$

Three arrays are employed to completely specify pollution control alternatives, their costs, and their pollutant output rates. These arrays are RALT, CALT, P1 and AP1; where

RALT(ID,I,J,K) = value of MS for pollution control alternative ID at source (I,J,K);

CALT(ID,I,J,K) = annual cost in dollars for pollution control alternative ID at source (I,J,K);

P1(I,J,K) = instantaneous pollutant output rate in kilograms per hour for pollution control alternative 1; i.e., ID = 1; and

AP1(I,J,K) = annual pollutant output in kilograms for pollution control alternative 1.

Note that the pollutant output rate for pollution control alternative 2; i.e., ID = 2, is zero kilograms per hour and that the pollution output rate in the absence of pollution control action is P(I,J,K) kilograms per hour. Values for the above arrays can be obtained by the relations shown below:

$$\text{If} \quad C(1,I,J,K) < VC \cdot (AP(I,J,K) + L(APA(I,J,K))),$$

$$\begin{aligned}
\text{then} \quad RALT(2,I,J,K) &= 3 \\
CALT(2,I,J,K) &= C(1,I,J,K) + C(2,I,J,K) - VC \cdot L(-APA(I,J,K)) \\
RALT(1,I,J,K) &= 1 \\
CALT(1,I,J,K) &= C(1,I,J,K) \\
P1(I,J,K) &= PA(I,J,K) \\
AP1(I,J,K) &= APA(I,J,K) .
\end{aligned}$$

$$\text{If} \quad C(1,I,J,K) \geq C(2,I,J,K) + VC \cdot AP(I,J,K) \ ,$$

$$\begin{aligned}
\text{then} \quad RALT(1,I,J,K) &= RALT(2,I,J,K) = 2 \\
CALT(1,I,J,K) &= CALT(2,I,J,K) = C(2,I,J,K) + VC \cdot AP(I,J,K) \\
P1(I,J,K) &= 0.0 \\
AP1(I,J,K) &= 0.0 .
\end{aligned}$$

Otherwise,

$$\begin{aligned}
RALT(2,I,J,K) &= 2 \\
CALT(2,I,J,K) &= C(2,I,J,K) + VC \cdot AP(I,J,K) \\
RALT(1,I,J,K) &= 1 \\
CALT(1,I,J,K) &= C(1,I,J,K) \\
P1(I,J,K) &= PA(I,J,K) \\
AP1(I,J,K) &= APA(I,J,K) . \qquad (C.22)
\end{aligned}$$

In addition to reducing the alternatives to be considered at a mine source, another case exists where the alternatives upstream of an active instream processor should be limited. Recall that abatement costs might be so inexpensive that treatment at a mine source should always be coupled with abatement. For the same reason, sources with inexpensive abatement costs providing influent to an active instream processor should involve abatement (or treatment and abatement) to achieve minimum system cost. In this case, no pollution control would be more expensive than abatement, but treatment and abatement may be implemented to achieve better stream quality. This situation is identified by the array BS, where

$$BS(I,J,K) = \begin{cases} \text{-1 when RALT(2,I,J,K) = 3 and the source at} \\ \text{(I,J,K) provides influent to an active instream} \\ \text{processor, and} \\ \\ \text{0 if otherwise.} \end{cases}$$

Modification of the Criterion Function and Constraint Equations -

Recall that the decisions at each instream treatment site are specified by the DINT array as defined on page 203; however, the decision alternatives at each mine source have been redefined in the previous section. The decision variable ID specifies the decision at each node, but the significance of this variable must be checked to insure that it conforms to the Lawler-Bell algorithm.

A requirement of the Lawler-Bell algorithm is that the criterion function is to be minimized, and it must be nondecreasing in each of the decision variables $x_1,x_2,\ldots,x_n$. Before the source and instream treatment decisions are reordered to correspond to elements of the vector X, the maximum effectiveness optimization problem must be transformed to satisfy the above requirement. If the criterion function; i.e., equation C.11, is multiplied by -1, then the maximization problem is transformed to a minimization problem. Recognition of this "trick" implies that the Lawler-Bell algorithm will work for a maximization problem where the criterion function is a nonincreasing function of each decision variable.

However, the criterion function as defined in equation C.11 is an increasing function of the decision variable at each mine source noted by ID as defined above. Consequently, the decision at each mine source is specified by

$$D(I,J,K) = 2\text{-}ID = \begin{cases} \text{2 if no pollution control is to be implemented at source (I,J,K)} \\ \\ \text{1 if the lowest cost pollution control alternative (usually abatement) is implemented at (I,J,K), and} \\ \\ \text{0 if the pollution control alternative involving treatment is implemented at (I,J,K)} \end{cases}$$

Note that the criterion function will be a nonincreasing function of each element in the D array and that it already is a nonincreasing function of the DINT array elements, defined on page 203, specifying decisions at each instream treatment site.

To incorporate the newly defined decision variables in the criterion function, equation C.16 giving the pollutant output rate from source (I,J,K) must be rewritten. Using the definitions given in equation C.22,

$$PS(I,J,K) = \begin{cases} P(I,J,K) \text{ if } D(I,J,K) = 2 \\ P1(I,J,K) \text{ if } D(I,J,K) = 1 \\ 0.0 \text{ if } D(I,J,K) = 0 \end{cases} \qquad (C.23)$$

Thus, the criterion function is evaluated using equations C.22, C.23, C.15, C.13, C.14, C.12, and C.11. The optimization program uses Function EFFECT to evaluate the criterion function, and the procedure for EFFECT is described in Figure C.8 starting on page 303.

The cost constraint is modified in a similar manner. That is, the cost to control pollution at a mine source given by equation C.19 is replaced by

$$CS(I,J,K) = \begin{cases} CALT(2,I,J,K) \text{ if } D(I,J,K) = 0 \\ CALT(1,I,J,K) \text{ if } D(I,J,K) = 1 \\ 0 \text{ if } D(I,J,K) = 2 \end{cases} \qquad (C.24)$$

In addition, the annual pollution output from a mine source given by equation C.21 becomes

$$APS(I,J,K) = \begin{cases} AP(I,J,K) \text{ if } D(I,J,K) = 2, \\ AP1(I,J,K) \text{ if } D(I,J,K) = 1, \text{ and} \\ 0.0 \text{ if } D(I,J,K) = 0 \end{cases} \qquad (C.25)$$

Hence, the cost constraint is evaluated using equations C.22, C.24, C.25, C.18, C.19, and C.20.

Construction of the Solution Vector X and Identification of the Next Vector to be Evaluated -

Having defined the decisions variables, the next step in developing the DWME optimization algorithm is to define a solution vector X consisting of these decision variables. Note that the decision variable at each mine source can assume three possible values; thus, the solution vector X consists of both binary and tertiary elements necessitating a new procedure for specifying the next vector, i.e., X*, to be evaluated. Both the solution vector X and the procedure for identifying the next vector to be evaluated are defined in this section.

As specified before, the vector X has n elements or

$$X = x_1, x_2, \ldots, x_m, x_{m+1}, \ldots, x_n \;;$$

where n = number of instream processors plus the number of mine sources.

That is,

$$n = \text{NINST} + \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} ND(J,K) \;.$$

x_m represents the decision at an instream processor or the decision at a mine source. Hence,

$$x_m = D(I,J,K)$$

or

$$x_m = \text{DINT}(NT)$$

These decision variables are assigned as elements of the vector using the following rules:

1. If $x_m = \text{DINT}(NT)$, then $x_{m+1} = D(I,J,K)$; where $NT = \text{INT}(I,J,K)$. That is, the instream processor decision variables are placed to the left of their corresponding mine source nodes.
2. The decision variables for each stream always appear together and ordered so that x_{m+1} is upstream of x_m assuming x_m represents the decision at a mine source.

3. If x_m represents the most upstream node of stream J of level K and $J > 1$, then x_{m+1} represents the most downstream node of stream J-1 of level K.

4. If x_m represents the most upstream node of stream 1 of level K and $K > KL$, then x_{m+1} represents the most downstream node of stream NS(K-1) of level K-1.

The above rules are designed to order these decision variables so that the downstream decisions are recorded to the left of upstream decisions.

To illustrate the application of the above rules, consider a basin consisting of one level-three stream, two level-two streams, and two level-one streams. Each stream has two nodes, and the downstream nodes of each level-two stream have potential instream processor sites. Then

$$X = [D(2,1,3),\ D(1,1,3),\ DINT(1),\ D(2,2,2),\ D(1,2,2,),\ DINT(2),\ D(2,1,2),$$
$$D(1,1,2),\ D(2,2,1),\ D(1,2,1),\ D(2,1,1),\ D(1,1,1)]\ .$$

The Lawler-Bell algorithm examines vectors in sequence by ordering each possible solution vector and working from the first to the last solution vector. This order is achieved by assigning a number to each solution vector and placing the vectors in numerical order. This numerical value for a solution vector is given by

$n(X)$ so

X is evaluted before Y if $n(X) < n(Y)$.

As specified by Lawler-Bell, the elements of the solution vector must be binary, thus $n(X)$ is just a binary number. For the DWME optimization model, $n(X)$ must give a numerical value to X where some elements are binary and some elements are tertiary. Thus,

$$n(X) = \sum_{m=1}^{n} x_m \cdot n_m\ ;$$

where $n_m = 2^{b_m} \cdot 3^{n-m-b_m}$, and

b_m = number of binary elements to the right of x_m, i.e., $x_{m+1}, s_{m+2}, \dots, x_n$

Note that the function $n(X)$ assigns integral values to the possible vectors X ranging from 0 to n_0. To illustrate the numerical ordering provided by $n(X)$, consider a system composed of a single stream having two nodes, and the upstream node has a potential instream treatment site. Thus,

$$x_1 = 0, 1, \text{ or } 2$$

$$x_2 = 0 \text{ or } 1$$

$$x_3 = 0, 1, \text{ or } 2$$

A listing of each possible vector in numerical order is shown below:

X	n(X)	X	n(X)
(0,0,0)	0	(1,1,1)	10
(0,0,1)	1	(1,1,2)	11
(0,0,2)	2	(2,0,0)	12
(0,1,0)	3	(2,0,1)	13
(0,1,1)	4	(2,0,2)	14
(0,1,2)	5	(2,1,0)	15
(1,0,0)	6	(2,1,1)	16
(1,0,1)	7	(2,1,2)	17
(1,0,2)	8		
(1,1,0)	9		

In the optimizing algorithm, we must be able to take an arbitrary vector X and find the next vector in the numerical ordering. That is, we must be able to regard X as a number, i.e., the function $n(X)$, and add one to X giving Y so that $n(Y) = n(X) + 1$. The following procedure is used to add one to X:

1. Starting from the rightmost element of X, i.e., x_n, find the first element noted as x_a where x_a is less than its maximum possible value. That is, $x_a < 2$ if x_a is tertiary and $x_a = 0$ if x_a is binary.
2. Add one to x_a, or $y_a = x_a + 1$.
3. Set $y_{a+1}, y_{a+1}, \ldots, y_n$ all equal to zero.
   Set $y_1 = x_1$, $y_2 = x_2, \ldots, y_{a-1} = x_{a-1}$.

The proof that the above procedure always gives $n(Y) = n(X)+1$ is done by induction on a. If x_a is the rightmost element, then it is obvious that $n(Y) = n(X)+1$. If $a = n-1$, then $y_{n-1} = x_{n-1} + 1$, $y_n = 0$, and x_n

equals its maximum possible value. Since

$$n(X) + 1 = \sum_{m=1}^{n-2} x_m \cdot n_m + x_{n-1} \cdot n_{n-1} + x_n + 1$$

and $n_{n-1} = x_n + 1$, then

$$n(X) + 1 = \sum_{m=1}^{n-2} x_m \cdot n_m + \left(x_{n-1} + 1\right) \cdot n_{n-1} \quad \text{and}$$

$$n(Y) = n(X) + 1 .$$

To show $n(Y) = n(X) + 1$ for all values of a, assume that $n(Y) = n(X) + 1$ for a and then show that this assumption implies $n(Y) = n(X) + 1$ for a-1. Thus we assume that

$$n(X) + 1 = \sum_{m=1}^{a-1} x_m \cdot n_m + x_a \cdot n_a + \sum_{m=a+1}^{n} x_m \cdot n_m + 1$$

$$= \sum_{m=1}^{a-1} x_m \cdot n_m + (x_a + 1)n_a$$

$$= n(Y)$$

when x_a is less than its maximum value and $x_{a+1}, x_{a+2}, \ldots, x_n$ are all equal to their maximum values. Note that the above expression must also be valid for the case where x_a is equal to its maximum value if it is true when x_a is less than its maximum value. For the a-1 case assume that $x_a, x_{a+1}, \ldots, x_n$ are all equal to their maximum values and x_{a-1} is not. Expanding the expression for n(X),

$$n(X) + 1 = \sum_{m=1}^{a-2} x_m \cdot n_m + x_{a-1} \cdot n_{a-1} + \sum_{m=a}^{n} x_m \cdot n_m + 1 .$$

Based on our assumed relationship for the a case,

$$n(X) + 1 = \sum_{m=1}^{a-2} x_m \cdot n_m + x_{a-1} \cdot n_{a-1} + (x_a + 1)n_a \,.$$

Note that $(x_a+1) \cdot n_a = n_{a-1}$ because x_a is equal to its maximum value. Thus,

$$n(X) + 1 = \sum_{m=1}^{a-2} x_m \cdot n_m + (x_{a-1} + 1)n_{a-1} \,,$$

which proves that the procedure for adding one to X is valid.

This procedure for adding one to X to find the succeeding vector in the numerical ordering must be extended to handle two additional cases. These cases occur when x_m represents a mine source at (I,J,K), and they are:

1. If RALT(1,I,J,K) = 2, then the decisions $x_m = 0$ and $x_m = 1$ are identical since abatement is more expensive than treatment. Thus, the decision $x_m = 1$ may be skipped.
2. If BS(I,J,K) = -1, then the decision represented by $x_m = 2$ is not permitted since an instream treatment processor downstream of (I,J,K) is active and no pollution control at (I,J,K) is more expensive than abatement.

The extended procedure incorporating the above cases for proceeding from X to Y, the next vector in the numerical ordering, is shown below:

1. Set m = n.
2. Set (I,J,K) = node coordinates for x_m.
   If $x_m = 2$, go to step 5.
   If $x_m = 1$, x_m represents a mine source, and BS(I,J,K) = -1, go to step 5.
   If $x_m = 1$ and x_m represents an instream processor, go to step 5.

3. x_m is less than its maximum value.
   Set $y_m = x_m + 1$.
   If $y_m = 1$, x_m represents a mine source, and RALT(1,I,J,K) = 2, set $y_m = 2$.

4. Go to step 6.

5. x_m is equal to its maximum value.
   Set $y_m = 0$.
   $m - 1 \rightarrow m$. (The operation $a + b \rightarrow a$ means add b to a and record the sum as the new value of a.)
   Go to step 2.

6. Set $y_1 = x_1, y_2 = x_2, \ldots, y_{m-1} = x_{m-1}$. The next vector in the numerical ordering has been determined.

Regard x_0, if encountered in the above procedure, as equal to zero.

In addition to the numerical ordering, the vector partial ordering is important since $X \leq Y$ implies that Y cannot have greater effectiveness than X. The vector partial ordering is defined by

$$X \leq Y \text{ if and only if } x_m \leq y_m \text{ for } m = 1,2,\ldots,n .$$

For example, if $X = (1,1,2)$ and $Y = (2,0,0)$, then $X \not\leq Y$ and $Y \not\leq X$, but $n(X) < n(Y)$.

As before, let X^* denote the first vector following X in the numerical ordering where

$$X \not\leq X^* .$$

The basic procedure for determining X^* is outlined below:

1. Starting from the rightmost element of X, i.e., x_n, find the first nonzero element. Designate this element as x_a.

2. Set x_a^* equal to the maximum value of x_a. Also, set all elements to the right of x_a^* equal to the maximum values of their respective elements. Set $x_m^* = x_m$ for $m = 1,2,\ldots,a-1$.

3. Add one to X^*.

Note that the vector X* obtained at the completion of step 2 above satisfies

$$X \leq X^* .$$

This is true since there is no element of X* (as determined by step 2) which proceeded past its maximum value in the numerical ordering between X and X*. By adding one to X* in step three, x_a^* becomes zero and smaller than x_a. Thus $X \nleq X^*$.

More explicitly, the procedure for determining X* is outlined below:

1. Set m = n.

2. Set (I,J,K) = node coordinates for the decision variable x_m. If $x_m > 0$, go to step 5.

3. Set $x_m^* = 2$ if x_m represents a mine source and $BS(I,J,K) \neq -1$. Otherwise, set $x_m^* = 1$.

4. $m - 1 \rightarrow m$.
   Go to step 2.

5. If x_m represents a mine source and $BS(I,J,K) \neq -1$, set $x_m^* = 2$.
   Set $x_b^* = x_b$ for $b = 1,2,\ldots,m-1$.

6. Replace X* with the next vector in the numerical ordering subsequent to X*.

7. The procedure is complete, i.e., $X \nleq X^*$.

Several examples are shown below to illustrate the procedure.

Example 1:

$X = (0,1,1,0,2,1,0)$,

where x_3 represents an instream treatment facility.

Then $X^* = (0,1,1,1,0,0,0)$.

Example 2:

$X = (0,1,1,1,0,0,0)$,

where x_3 represents an instream treatment facility.

Then X* = (0,2,0,0,0,0,0).

Example 3:

X = (0,1,1,1,2,1,0),

where x_4 represents the mine source at node (2,1,3), RALT(2,2,1,3) = 3,
BS(2,1,3) = -1,

x_3 represents the mine source at node (3,1,3), and BS(3,1,3) = -1.

Then X* = (0,2,0,0,0,0,0).

Example 4:

X = (0,1,1,2,0,0,0),

where x_3 represents the mine source at node (3,1,3) and BS(3,1,3) = -1.

Then X* = (0,2,0,0,0,0,0).

Determination of the Next Feasible Solution -

The value of X* obtained in the procedure described above may be more effective than X, but X* may not be a feasible solution since it could violate the cost constraint. At this point in the Lawler-Bell algorithm, a check is made to determine whether X* is feasible. If so, the procedure outlined in extension 3 above is repeated. Otherwise, checks are made to determine if more vectors could be skipped to find a feasible solution. This search for a feasible solution would involve many iterations.

A much more efficient method is used by the DWME algorithm where a procedure has been developed which identifies precisely the next feasible solution in the numerical ordering beyond an arbitrarily selected point. Since this procedure does not require much more computational effort than to evaluate the cost constraint, the overall efficiency of the optimization algorithm has been clearly enhanced. Subroutine NEFESE is used to specify the next feasible solution. The method employed by NEFESE is outlined in this section, and a flowchart of the subroutine is presented in Figure C.10 on page 309.

The basic idea inherent in the procedure to identify the next feasible solution makes use of the lexicographic nature of the numerical ordering of possible values for X. The numerical ordering given by n(X) is lexicographic because addition of one to an element, e.g., x_a, increases the numerical value of X by more than the possible contribution of all elements to the right of x_a. This is true because

$$(x_a + 1)n_a = 1 + \sum_{m=a}^{n} x_m \cdot n_m ,$$

where $x_{a+1}, x_{a+2}, \ldots, x_n$ all have their maximum possible values.

Because of this lexicographic property, the next feasible solution procedure can start with the leftmost element, i.e., x_1, determine its value for the next feasible solution and then do likewise for elements on the right. To specify this basic procedure, let

$X = (x_1, x_2, \ldots, x_n)$ = the current solution (not necessarily feasible)

$X^* = (x_1^*, x_2^*, \ldots, x_n^*)$ = the next feasible solution vector equal to or after X in the numerical ordering.

Note that $X^* = X$ if X is feasible. Consider the determination of x_1^* to illustrate the procedure. Compute the cost of the decision x_1 under the assumption of no pollution control costs to the right of x_1, and note this cost as TTC. If TTC is less than BUD, the maximum allowable pollution control expenditure, then allocate TTC out of BUD and set $x_1^* = x_1$. If TTC is greater than BUD, then x_1^* must be greater than x_1 to achieve a feasible solution. If x_1^* is forced to be greater than x_1, no intervening feasible solutions between X and X* are skipped if $x_2, x_3, \ldots, x_n$ are all set to zero. Setting x_1^* to be greater than x_1 if required is permitted because of the lexicographic nature of the numerical ordering and the assumption that no pollution costs are incurred as a result of decisions to the right of x_1. After x_1^* is determined, x_2^* is determined, but now the allowable budget is BUD - TTC. Again, the lexicographic nature of the numerical ordering permits a serial allocation of available budget in this manner.

The calculation of TTC when x_1 is a decision variable that represents a mine source is straightforward; however, instream treatment processor decisions are complicated by the fact that upstream pollution control decisions change treatment variable costs. Increasing pollution control upstream will always decrease treatment variable costs; however, most upstream pollution actions will increase total system cost. An exception is made when upstream abatement will decrease total cost, and this situation is identified when BS(I,J,K) = -1. The principle that

TTC will represent the minimal increase in total system cost will be invoked to calculate TTC for an instream processor decision. Consequently, the cost TTC when x_1 represents a treatment processor consists of the sum of the following components:

1. CI(NT), the instream treatment processor fixed cost.
2. VC · APINS(NT), where APINS(NT) is the annual pollutant flow past instream processor NT under the assumption the only upstream pollution control actions taken will be when abatement reduces total system cost, i.e., BS(I,J,K) = -1.
3. The total cost to perform abatement at each upstream node where BS(I,J,K) = -1. This cost is computed by Function CABAT (see Figure C.6 on page 296 for flow chart of CABAT).

At upstream nodes, the calculation of costs to determine values of x_m^* must consider interactions with active downstream treatment processors. Two arrays are used to determine if a decision variable is upstream of an active downstream treatment processor. These arrays are:

$$\text{KNINT(I,J,K)} = \begin{cases} \text{1 if mine source at (I,J, K) is upstream of an active treatment processor,} \\ \text{0 if otherwise} \end{cases}$$

$$\text{KSINT(NT)} = \begin{cases} \text{1 if the instream processor site NT is upstream of another active instream processor, and} \\ \text{0 if otherwise} \end{cases}$$

These arrays are maintained by subroutine TONE whenever an instream processor is initiated and subroutine TOFE when an instream processor is deactivated. Moreover, these subroutines maintain the array BS to specify when abatement or treatment and abatement is required at upstream nodes. See Figures C.11 and C.12 for flowcharts of these routines. If a decision variable is upstream of an active instream processor, then function CSAVE can be used to determine the reduction in treatable annual pollutant flow at the downstream treatment site that can be realized by a particular decision at the upstream site. The value returned by CSAVE can be used in determining the savings in treatment costs by virtue of a pollution control decision at the upstream site. CSAVE uses an array APST to maintain a running balance of the annual pollution flow past each instream treatment site, where

APST(NT) = annual pollution flow in kilograms past treatment site NT based on the current values of X*.

The values for APST(NT) are initially set equal to APINS(NT), and then they are altered as values for X* are determined at upstream nodes. Of course, CSAVE only recognizes a reduction in treatable or positive values of APST(NT).

With the above mechanisms for handling interactions with instream processors, the computations for determining a value for x_m^* can be specified. Let TTC now represent the total allocated pollution control cost for decisions $x_1^*, x_2^*, \ldots, x_{m-1}^*$; thus, the cost calculations for x_m^* determine the increase in TTC such that TTC $\leq$ BUD and $x_m^* \geq x_m$. Also, let TC represent the trial total allocated pollution control cost for the decision represented by x_m^*, and TTC will become equal to TC if TC $\leq$ BUD.

If x_m represents the decision at instream processor NT, the following procedure is used to specify x_m^*:

1. If $x_m = 1$ (or DINT(NT) = 1), go to step 8.

2. Set (I,J,K) equal to the node coordinates for NT.

3. Compute the trial allocated cost, i.e.,
   TC = TTC + CI(NT) + APINS(NT) · VC + CABAT(I,J,K,IS) if
   APINS(NT) $\geq$ 0; otherwise
   TC = TTC + CI(NT) + CABAT(I,J,K,IS).
   (The variable IS is only used to improve the efficiency of the computer program and is not necessary to understand the basic computational procedure.)

4. If KSINT(NT) = 0, go to step 6.

5. Set SAVE equal to the savings in treatable annual pollution flow; i.e., SAVE = CSAVE(-1,I,J,K,IS,NX); where NX is determined by function CSAVE and is the active downstream site treating flow from NT.
   TC - VC · SAVE → TC.

6. If TC $\leq$ BUD, go to step 9.

7. The budget will not permit NT to remain active; thus,

   $$x_m^* = 1 \text{ or DINT(NT)} = 1$$

   CALL TOFE(NT,I,J,K)
   CALL ZOE(I,J,K)

Subroutine ZOE sets the decision variables $x_{m+1}, x_{m+2}, \ldots x_n$ to zero to avoid skipping any feasible solutions. The computations are complete.

8. $x_m^* = 1$ or DINT(NT) = 1
   The computations are complete.

9. The budget will permit NT to remain active.
   TTC = TC
   APST(NT) = APINS(NT)
   If KSINT(NT) = 1, set APST(NX) - SAVE → APST(NX).
   $x_m^* = 0$
   The computations are complete.

If x_m represents the decision at the mine source located at (I,J,K), the following procedure is used to specify x_m^*:

1. If $x_m = 2$, go to step 10.

2. If $x_m = 1$, go to step 6.

3. TC = TTC + CALT(2,I,J,K)
   If KNINT(J,J,K) = 0, go to step 5.

4. (I,J,K) is upstream of an active instream treatment facility. Set SAVE equal to the savings in annual pollution flow if treatment is implemented at this site.
   SAVE = CSAV(0,I,J,K,IS,NX)
   TC - VC · SAVE → TC
   If BS(I,J,K) = -1, TC - CALT(1,I,J,K) → TC.

5. If TC $\leq$ BUD, go to step 8.
   x_m^* will be at least one, so set $x_m = 1$.
   Call subroutine ZOE to set $x_{m+1}, x_{m+2}, \ldots, x_n$ equal to zero so that no feasible solutions are skipped.

6. If BS(I,J,K) = -1, go to step 10.
   TC = TTC + CALT(1,I,J,K)
   If KNINT(I,J,K) = 0, go to step 7.
   Compute the annual savings in pollution flow at the downstream processor; i.e., SAVE = CSAV(1,I,J,K,IS,NX)
   TC   SAVE · VC → TC.

7. If TC $\leq$ BUD, go to step 10.
   The budget will not permit pollution control action at this site.
   $x_m = 2$.
   Call subroutine ZOE to set $x_{m+1}, x_{m+2}, \ldots, x_n$ to zero.

Go to step 10.

8. Increase the resource allocation by
   TTC = TC
   If KNINT(I,J,K) = 0, go to step 10.

9. Update the annual pollution flow at the downstream treatment site
   APST(NX) - SAVE → APST(NX).

10. $x_m^* = x_m$.

Specification of the Last Decision Vector to be Evaluated -

In addition to providing for a more rapid method of searching the list of decision vectors, savings in computational effort can be gained by recognizing when further searching will not uncover a more effective solution. The Lawler-Bell algorithm requires that the entire list of decision vectors be examined; thus, the search can certainly terminate when x_0 becomes greater than zero or $n(X) = n_0$. Actually the search can terminate prior to this point, and the stop criterion is described below.

The last decision vector in the numerical sequence would be made up of maximum values for each decision variable. The physical significance of this vector is that no pollution control action would be taken, and the cost of this vector would be zero. Another decision vector exists in the numerical sequence, called X_L, where all vectors beyond X_L merely reduce control action specified by X_L without adding any new control action. If X_L is feasible, there is no need to evaluate decision vectors beyond X_L since the criterion function or system effectiveness will be nonincreasing at that point.

The stopping decision vector, X_L, can be determined by allocating the pollution control budget to the rightmost elements of the decision vector. All other elements would be set to their maximum values. It follows that decision vectors past X_L in the numerical sequence will only reduce the allocation specified by X_L. Let $X_L = \left(x_1^L, x_2^L, \ldots, x_n^L\right)$; and let X_L be made up of decision elements at mine sources, where

DL(I,J,K) = value of the stopping decision vector element at the mine source (I,J,K)

and decision elements at instream treatment processors, where

DINTL(NT) = value of the stopping decision vector at the instream treatment site NT.

The procedure for determining values of these elements is summarized below:

```
1.  Set TTC = 0.0.
    TTC will be used to accumulate the allocated budget.
    I = 1
    J = 1
    K = KL
    PUL = 0.0
    PUL will be used to accumulate annual natural pollutant
    flow.

2.  If K = KL, go to step 3.
    NF = JN(I,J,K-1)
    If NF ≤ 0, go to step 3.
    This is a confluence node.  Add in flow from the tributary.
    PUL + PTL(NF,K-1) → PUL, where
    PTL(J,K) = pollutant output from stream (J,K)

3.  Set TTC1 = TTC + CALT(2,I,J,K)
    If TTC1 > BUD, go to step 10.

4.  DL(I,J,K) = 0
    TTC = TTC1
    PUL + APN(I,J,K) → PUL
    NT = INT(I,J,K)
    If NT ≤ 0, go to step 6.

5.  This node has a potential instream treatment site.
    TTC1 = TTC + CI(NT) + VC · PUL if PUL ≥ 0; otherwise
    TTC1 = TTC + CI(NT)
    If TTC1 > BUD, go to step 10.
    DINTL(NT) = 0
    TTC = TTC1.

6.  If I ≥ ND(J,K), go to step 7.
    I + 1 → I
    Go to step 2.

7   PTL(J,K) = PUL
    PUL = 0.0
    If J ≥ NS(K), go to step 8.
    J + 1 → J
    I = 1
    Go to step 2.

8.  If K ≥ 3, go to step 9
    J = 1
    I = 1
```

$K + 1 \rightarrow K$
Go to step 2.

9. There is sufficient budget to implement each possible control action in the entire basin. Computations are complete.

10. The budget has been allocated. Set the remaining decision vector elements to their maximum values. Computations are complete.

Note that large pollution control budgets will give lower values of X_L in the numerical ordering. However, a large budget will also give lower values of X for the first feasible pollution. Thus, an interesting interplay exists between budget size and the set of solutions to be evaluated.

Overall Computational Procedure for DWME Model -

Program MAXEF determines the maximum effectiveness solution for the basin. MAXEF incorporates the extensions described above to the Lawler-Bell algorithm in determining the optimal solution. This optimal set of decisions is recorded in the two arrays whose elements are defined below:

O(I,J,K) = optimal value of D(I,J,K)

OINT(NT) = optimal value of DINT(NT).

To determine the optimal solution, the most effective or maximum value of the effectiveness function is recorded in the variable EF. As new feasible decision vectors are found with greater effectiveness than EF, EF is increased, and the decision vector is stored in the O and OINT arrays. Note that there is likely to be more than one solution that can give the same value for EF. As new feasible solutions are uncovered with effectiveness measures equal to EF, the lowest cost solution is retained in the O and OINT arrays. The cost for the solution currently recorded as optimal is recorded in the variable TC.

The procedure followed by MAXEF in determining an optimal solution is outlined below:

1. Initialize variables

   $TC = 10^{31}$

   $EF = -10^{31}$

   Set the D and DINT arrays to zero values.
   Compute the elements of the stopping vector, i.e., the values of the DL and DINTL arrays.

2. Call subroutine NEFESE to determine the next feasible solution which is recorded in the D and DINT arrays. NEFESE also sets TTC = cost of feasible solution given by D and DINT arrays.

3. Compute the effectiveness of this solution using function EFFECT, and record the effectiveness in EFF.

4. If EFF < EF, go to step 5.
   If EFF = EF and TTC > TC, go to step 5.
   Record a new optimal solution in the O and OINT arrays.
   EF = EFF
   TC = TTC.

5. Using the procedure outlined in extension 3, skip to the next decision vector in the numerical ordering which could have greater effectiveness than EF. Record this decision vector in the D and DINT arrays.

6. Check to determine whether the decision vector in the D and DINT arrays is past the decision vector made up of elements from the DL and DINTL arrays in the numerical ordering. If so, go to step 8.

7. Go to step 2.

8. The optimal solution is recorded in the O and OINT arrays. Its effectiveness is EF and cost is TC.

A description of program MAXEF appears in Figure C.5 starting on page 258.

Deterministic "Worst-Case" Minimum Cost (DWMC) Model -

In this section, the DWMC model is presented, and the optimization method for finding the minimum cost solution to achieve a fixed quality standard is described. A considerable amount of the notation and method for this optimization model is based on the notation and equations explained earlier for the DWME model. As much as possible, the same variable names are used for both optimization models to facilitate the understanding of both programs. Thus, references will be made to equations developed earlier.

A complete description of the model is obtained in three steps. First, the criterion function and constraint equations are specified to present the model. Next extensions to the Lawler-Bell algorithm are presented and outlined. The overall procedure used by program ALCOT is then described. Finally, computer program descriptions, input data instructions, and flow charts are contained in a later section of this appendix.

Criterion Function and Constraint Equations

The minimum cost model uses an expression for the total pollution control cost as its criterion function and expression for the allowable pollution concentration at each node as the constraint equations. The corresponding expressions developed for the maximum effectiveness model can be used with one significant modification. Recall that the partial enumeration algorithm requires a nondecreasing criterion function of each decision variable for a minimization problem. Since the decision variables, as defined for the DWME model, would decrease cost as they are increased, their meaning must be inverted for the minimum cost model. Hence, the following definitions are used for DWMC model:

$$\mathrm{DINT(NT)} = \begin{cases} 1 \text{ if instream processor NT is implemented where NT} > 0 \\ 0 \text{ if otherwise} \end{cases}$$

$$\mathrm{D(I,J,K)} = \mathrm{ID} = \begin{cases} 2 \text{ if the pollution control alternative involving treatment is implemented at (I,J,K)} \\ 1 \text{ if the lowest cost pollution control alternative (usually abatement) is performed at mine source (I,J,K), and} \\ 0 \text{ if no pollution control actions are taken at mine source (I,J,K)} \end{cases}$$

Incorporating the above definitions into the expression for total system cost; i.e., equation C.18, the criterion function for the DWMC model becomes

$$\mathrm{TTC} = \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} \sum_{I=1}^{ND(J,K)} \Big[\mathrm{CS(I,J,K)} + \mathrm{DINT(INT(I,J,K))}\cdot(\mathrm{CI(INT(I,J,K))} - \mathrm{VC}\cdot \mathrm{L(-APLT(I,J,K))})\Big] \qquad (C.26)$$

The annual cost of pollution control decisions at mine source (I,J,K), CS(I,J,K), is now specified by

$$\mathrm{CS(I,J,K)} = \begin{cases} \mathrm{CALT(2,I,J,K)} \text{ if } \mathrm{D(I,J,K)} = 2 \\ \mathrm{CALT(1,I,J,K)} \text{ if } \mathrm{D(I,J,K)} = 1 \\ 0 \text{ if } \mathrm{D(I,J,K)} = 0 \end{cases} \qquad (C.27)$$

Also, the annual pollutant load passing stream node (I,J,K) becomes

$$\begin{aligned} \mathrm{APLT}(I,J,K) = {} & (1-\mathrm{DINT}(\mathrm{INT}(I,J,K)))\cdot(\mathrm{APN}(I,J,K) + \mathrm{APLT}(I-1,J,K) \\ & + \mathrm{APS}(I,J,K) + \mathrm{APT}(\mathrm{JN}(I,J,K-1),K-1)) + \mathrm{DINT}(\mathrm{INT}(I,J,K)) \\ & \cdot L(\mathrm{APN}(I,J,K) + \mathrm{APLT}(I-1,J,K) + \mathrm{APS}(I,J,K) \\ & + \mathrm{APT}(\mathrm{JN}(I,J,K-1),K-1)) \qquad (C.28) \end{aligned}$$

The annual pollution output from a mine source is an input to the above equation and is specified

$$\mathrm{APS}(I,J,K) = \begin{cases} 0 \text{ if } D(I,J,K) = 2, \\ \mathrm{AP1}(I,J,K) \text{ if } D(I,J,K) = 1, \text{ and} \\ \mathrm{AP}(I,J,K) \text{ if } D(I,J,K) = 0. \end{cases} \qquad (C.29)$$

Equations C.26 through C.29 constitute the DWMC model criterion function which is to be minimized. Function TCOST is used by the DWMC model to compute values of system cost given the decision arrays D and DINT, and a flowchart at TCOST is shown in Figure C.22 on page 436.

The constraint equations for this DWMC model are derived from the objective of maintaining the maximum pollutant concentration below a specified level. Let

QS = maximum allowable pollution concentration in ppm.

Thus,

$$\frac{\mathrm{PLT}(I,J,K)}{\mathrm{PLT}(I,J,K) + Q(I,J,K)} \leq \mathrm{QS} \cdot 10^{-6}$$

$$\begin{aligned} \text{for } \; & I = 1,2,\ldots,\mathrm{ND}(J,K); \\ & J = 1,2,\ldots,\mathrm{NS}(K); \text{ and} \\ & K = \mathrm{KL},\ \mathrm{KL}+1,3 \qquad (C.30) \end{aligned}$$

Using the decision variable definitions specified above, in equation C.30, the pollution flow just downstream of node (I,J,K) is

$$\begin{aligned} PLT(I,J,K) = {} & (1-DINT(INT(I,J,K)))\cdot(PN(I,J,K) + PLT(I-1,J,K) + PS(I,J,K) \\ & + PT(JN(I,J,K-1),K-1)) + DINT(INT(I,J,K))\cdot L(PN(I,J,K) \\ & + PLT(I-1,J,K) + PS(I,J,K) + PT(JN(I,J,K-1),K-1)) \end{aligned} \tag{C.31}$$

$$\text{for } I = 1,2,\ldots,ND(J,K);$$
$$J = 1,2,\ldots,NS(K); \text{ and}$$
$$K = KL,\ KL + 1,3.$$

The pollution output rate from source (I,J,K), required for the above equation, is given by

$$PS(I,J,K) = \begin{cases} 0 \text{ if } D(I,J,K) = 2 \\ P1(I,J,K) \text{ if } D(I,J,K) = 1 \\ P(I,J,K) \text{ if } D(I,J,K) = 0. \end{cases}$$

The constraints for the DWMC model are given by equations C.30, C.31, and C.32.

Optimization Algorithm

The criterion function and constraint equations defined above are already in a suitable form for application of a partial enumeration optimization algorithm. The requirement for a nondecreasing criterion function of each decision variable in a minimization problem has been satisfied. Also, the rules for expressing the set of decision variables, i.e., the D and DINT arrays, as elements of a vector X can be applied directly from the DWME algorithm.

In a manner similar to the DWME algorithm, several extensions to the Lawler-Bell algorithm have been developed to substantially reduce computation effort. These extensions include a method for specifying the next feasible vector in the numerical ordering and a method for designating the stopping vector in the numerical sequence. Moreover, a method for decomposing the basin system into subsystems has been developed for the minimum cost algorithm. This decomposition permits substantial additional reductions in the number of iterations. The instream treatment facilities, when implemented, are natural points at which this decomposition is made.

Decomposition at Instream Processors

The interaction between decisions upstream of an active instream processor and the remainder of the basin is nil. In that case the pollutant output of the instream processor node is fixed at zero, and the decisions which minimize the criterion function are those which give a minimum cost solution upstream of the instream processor regardless of the decisions in the remainder of the basin. Consequently, once a minimum cost solution upstream of an instream processor has been found, this solution will be optimal whenever the instream processor is implemented.

Implementation of this decomposition at active instream processors is facilitated by the basic numerical order, or lexicographic order, in which the possible solutions are enumerated. That is, once an instream processor is implemented, its decision variable x_m in the decision vector X is changed from zero to one; and all decisions to the right of x_m, i.e., $x_{m+1}, x_{m+2}, \ldots, x_n$, must cycle completely through their numerical sequence from each element having a value of zero to each element being equal to its maximum value. More explicitly, once the instream processor is activated, then

1. Its decision variable x_m in the decision vector X is changed from zero to one;

2. All decision variables to the right of x_m must cycle through all alternatives from

   $$\left(x_{m+1} = 0,\ x_{m+2} = 0; \ldots, x_n = 0\right) \text{ to}$$

   $$\left(x_{m+1} = x^m_{m+1},\ x_{m+2} = x^m_{m+2}, \ldots, x_n{}^m\right)$$

   where $x_a{}^m$ = maximum value of decision vector element a

   $x_a{}^m = 1,2;$

3. The remainder of the decision vector to the left of x_m is held constant until

   $$\left(x_{m+1} = x^m_{m+1},\ x_{m+2} = x^m_{m+2}, \ldots, x_n{}^m\right)$$

   is reached.

The least cost feasible solution occurring while the numerical sequence from $\left(x_m = 0,\ x_{m+1} = 0,\ x_n = 0\right)$ to $\left(x_{m+1}, x^m_{m+2}, \ldots, x_n{}^m\right)$ is being searched is an optimal upstream solution to the instream processor. This is true because the vector $\left(x_{m+1}, x_{m+2}, \ldots, x_n\right)$ contains all the nodes upstream of x_m. Although this vector may have nodes in addition to those upstream of x_m, all possible combinations of those nodes upstream of x_m and those not upstream of x_m but to the right of x_m are searched.

To capitalize on this decomposition, the first time that an instream processor is activated, the optimal upstream solution is determined as each upstream decision vector is searched. Then the next time the instream processor is activated, its optimal upstream solution is immediately implemented and used until the processor is deactivated.

The mechanism for implementing this decomposition by program ALCOT is described below. Four variables are used in the process of determining the optimal upstream solution, and they are:

$$OI(L) = \begin{cases} 1 \text{ if instream processor L has had an optimal upstream solution calculated} \\ 0 \text{ if otherwise,} \end{cases}$$

IO(I,J,K,L) = optimal value of D array for source (I,J,K) and upstream solution to the instream processor L,

IOI(NT,L) = optimal value of DINT array for instream processor NT and upstream solution to instream processor L,

CIST(L) = total cost of solution recorded in IO and IOI arrays for upstream solution to instream processor L;

where L = 1,2,...,NINST;

I = 1,2,...,ND(J,K);

J = 1,2,...,NS(K);

K = KL, KL+1, 3; and

NT = 1,2,...,NINST.

When instream processor L is activated, subroutine TON is called to prepare upstream variables. TON can determine whether L has an optimal upstream solution calculated by the value of OI(L). The first time the processor is activated, TON sets BS(I,J,K) = -1 for each upstream node where abatement is cheaper than no pollution control due to savings in treatment variable costs. Also, TON initializes CIST(L) to a large number. As program ALCOT determines feasible basin solutions having lower cost than CIST(L), then these solutions are stored in the IO and IOI arrays; and the value of CIST(L) is adjusted. Later, when ALCOT progresses sufficiently through the numerical sequence of the solution vector to deactivate instream processor L, then ALCOT calls subroutine

TOFF to record the fact that the upstream solution for L in the IO and IOI arrays is optimal by setting OI(L) = 1. Also, TOFF resets

$$BS(I,J,K) = 0$$

$$D(I,J,K) = 0$$

for any upstream node where

$$BS(I,J,K) = -1.$$

Once the above procedure has been completed, subsequent activations of the instream processor will be accompanied by immediate jumps to the optimal upstream solution. Two additional arrays are employed by program ALCOT to freeze the upstream solution at its optimal value. These variables are:

$$BS(I,J,K) = \begin{cases} 1 \text{ if the decision variable for source } (I,J,K) \text{ is not being varied because it is frozen as part of an optimal upstream solution,} \\ 0 \text{ if no restrictions are being placed on the decision variable for source } (I,J,K)\text{, and} \\ -1 \text{ if abatement or treatment and abatement must be performed at source } (I,J,K) \text{ because it is upstream of an active instream processor and no control is more expensive than abatement.} \end{cases}$$

$$BT(NT) = \begin{cases} 1 \text{ if the decision variable for instream processor NT is frozen as part of an optimal upstream solution, and} \\ 0 \text{ if no restrictions are being placed on the decision variable for instream processor NT.} \end{cases}$$

If subroutine TON is called after instream processor NT is activated and an optimal upstream solution has been calculated, i.e., OI(NT) = 1, then TON sets the values in the D and DINT arrays upstream of NT to the optimal values recorded in the IO and IOI arrays. In addition, each node upstream of NT has its values in the BS and BT arrays set to one. The BS and BT arrays prevent any changes to elements of the D and DINT arrays upstream of NT. Later, when NT is deactivated, the entries in the D, DINT, BS, and BT arrays upstream of NT are set to zero.

The procedures noted above to take advantage of possible decompositions at active instream processor sites necessitate changes in the procedure to skip solutions that are obviously nonoptimal. Recall that a procedure is specified on page 219 for skipping from an arbitrary vector X to a vector X* which is the first vector subsequent to X in the numerical ordering which could be optimal. That is, X* is the first vector subsequent to X in the numerical ordering where

$$X \not\leq X^*.$$

This procedure requires another procedure to determine the next vector in the numerical ordering in which blocked and redundant solutions are skipped. The amended procedure for finding Y the next vector in the numerical ordering subsequent to X is outlined below:

```
1. Set $m = n$.

2. If $x_m$ represents an instream treatment facility, go to step 7.

3. Set (I,J,K) = node coordinates for $x_m$.
   If BS(I,J,K) = 1, go to step 6.
   If $x_m = 2$, go to step 5.

4. $x_m$ is less than its maximum value.
   Set $y_m = x_m + 1$.
   If $y_m = 1$ and RALT(1,I,J,K) = 2, set $y_m = 2$.
   Go to step 10.

5. $x_m$ is equal to its maximum value.
   If BS(I,J,K) = -1, set $y_m = 1$; otherwise set $y_m = 0$.
   $m - 1 \rightarrow m$.
   Go to step 2.

6. $x_m$ is frozen at its current value.
   Set $y_m = x_m$.
   $m - 1 \rightarrow m$
   Go to step 2.

7. Set NT = the instream treatment number for $x_m$.
   If BT(NT) = 1, go to step 6.
   If $x_m = 1$, go to step 9.

8. $x_m$ is less than its maximum value.
   Set $x_m = 1$.
   Go to step 10.
```

9. x_m is equal to its maximum value.
   Set $x_m = 0$.
   $m - 1 \rightarrow m$
   Go to step 2.

10. Set $y_1 = x_1, y_2 = x_2, \ldots, y_{m-1} = x_{m-1}$.
    The next vector in the numerical ordering has been determined.

The procedure for determining X*, which is the first vector subsequent to X in the numerical ordering that could be optimal, is listed below.

1. Set $m = n$.

2. If x_m represents an instream treatment site, go to step 7.

3. Set (I,J,K) = node coordinates for the decision variable x_m.
   If BS(I,J,K) = 1, go to step 6.
   If BS(I,J,K) = -1 and $x_m = 1$, go to step 5.
   If $x_m = 0$, go to step 5.

4. x_m is greater than its minimum value.
   Set $x_m^* = 2$.
   Go to step 10.

5. x_m equals its minimum value.
   Set $x_m^* = 2$.
   $m - 1 \rightarrow m$
   Go to step 2.

6. x_m is frozen at its current value.
   Set $x_m^* = x_m$.
   $m - 1 \rightarrow m$
   Go to step 2.

7. Set NT = the instream treatment site number for decision variable x_m.
   If BT(NT) = 1, go to step 6.
   If $x_m = 1$, go to step 9.

8. x_m is equal to its minimum value.
   Set $x_m^* = 1$.
   $m - 1 \rightarrow m$
   Go to step 2.

9. x_m is greater than its minimum value.
   Set $x_m^* = 1$.

10. Set $x_b^* = x_b$ for $b = 1,2,\ldots,m-1$.
    Replace X* with the next vector in the numerical ordering subsequent to X*.

Determination of the Next Feasible Solution -

For the same reasons as implemented in the DWME algorithm, a procedure has been developed for the DWMC algorithm to determine the next feasible solution in the numerical ordering given an arbitrary decision vector X. A considerable number of iterations can be saved by proceeding directly to the next feasible solution in one step. Let,

$X = x_1,x_2,\ldots,x_n$ = the current solution (not necessarily feasible); and

$X^* = x_1^*,x_2^*,\ldots,x_n^*$ = the next feasible solution vector equal to or after X in the numerical ordering

The vector X* is determined in the algorithm by subroutine NEXFES (see Figure C.17 for a flowchart of NEXFES).

The procedure for determining the next feasible solution relies heavily upon the lexicographic nature of the numerical ordering and is illustrated by considering the problem of determining x_1^*. Let QMAX be the maximum pollutant that can be emitted from the node represented by x_1 without violating the quality constraint. In calculating QMAX, assume that treatment is being performed at all upstream nodes. If the decision implied by x_1 emits pollutant at a rate less than or equal to QMAX, then x_1^* must equal x_1 because intervening feasible solutions in the numerical ordering would be skipped otherwise. For example, the solution with treatment being applied at each upstream node would be skipped if $x_1^* > x_1$. The other cases occur when the decision implied by x_1 emits more pollutant than QMAX. When this occurs, x_1^* must be greater than x_1, but x_1^* is set to the smallest value such that the pollutant output is less than or equal to QMAX. Also, if x_1^* is forced to be greater than x_1, then $x_2,x_3,\ldots,x_n$ are all set to zero for subsequent computation in the procedure. Setting these variables to the right of x_1 to zero when x_1^* is greater than x_1 is required because of the lexicographic nature of the numerical ordering and to avoid skipping intervening feasible solutions. Subroutine ZO, flowcharted in Figure C.25, is used to set decision variables to zero.

After x_1^* is determined, then x_2^* is determined in the same manner but based upon the value of x_1^*. Thus, QMAX must be calculated considering the quality standards at the nodes for x_2 and x_1 and the node for x_1 receives pollutant at the rate specified by x_1^*. The lexicographic nature of the numerical ordering permits this sequential solution process.

A minor change in the previously defined variable PN(I,J,K) simplifies calculations performed by subroutine NEXFES considerably. Recall that the calculation of QMAX for a specified node is based upon all upstream decision variables having treatment specified. Then the only pollutant input to the specified node would be natural pollutant that was not removed by instream treatment processes. Let this natural pollutant input be

PN(I,J,K) = natural pollutant input to node (I,J,K) in kilograms per hour from all upstream sources assuming all instream treatment processes are activated.

Note that this definition implies that PN(I,J,K) is the natural pollutant flow just upstream of (I,J,K). The definition of PN(I,J,K) has been changed solely for the convenience of the program and all subsequent references to PN(I,J,K) will conform to the revised definition unless input data formats are being discussed. Natural pollutant input data values will be the incremental inputs between nodes as defined previously.

Using the above definition, the value of QMAX can be readily calculated. QMAX represents the maximum input from a specified node that can be tolerated at the node and at all downstream nodes. Thus, a relationship for determining the maximum additional pollutant flow that can be accepted at a node without violating the quality standard is used. Let PLT(I,J,K) be the total pollutant flow including natural pollutant that exists just downstream of node (I,J,K) and let PM(I,J,K) be the maximum additional pollutant flow that can be accepted at the node. Using these variables and Q(I,J,K), one obtains the equation,

$$\frac{PLT(I,J,K) + PM(I,J,K)}{PLT(I,J,K) + PM(I,J,K) + Q(I,J,K)} = QS \cdot 10^{-6} \text{ or}$$

$$PM(I,J,K) = \frac{QS \cdot 10^{-6}}{1 - QS \cdot 10^{-6}} \cdot Q(I,J,K) - PLT(I,J,K), \tag{C.33}$$

if the node does not have an active instream treatment facility. With an instream treatment facility, PM(I,J,K) can be as large as desired.

As outlined earlier, the procedure for determining the next feasible solution involves a recursive procedure starting with x_1* and working toward x_n*. The method for performing one step in the recursive procedure; given the results of computations defining $x_1*, x_2*, \ldots, x_{m-1}*$; is outlined below. The procedure operates more efficiently when x_m represents an instream processor if both x_m* and x_{m+1}* are determined in the same recursive, step. In this case, note that x_{m+1}* will always represent the mine source decision for the same node as x_m*.

```
1.  Set (I,J,K) equal to the node coordinate for $x_m$
    NT = INT(I,J,K)
    PC = PN(I,J,K)
    PE = additional pollutant that could be released if an
         instream treatment facility were not used.
    PE = 0.0
    If NT = 0, go to step 3.

2.  $x_m$ represents an instream treatment facility.
    If PC $\leq$ 0, go to step 3.
    If $x_m$ = 1, PC = 0.0
    If $x_m$ = 0, PE = PC.

3.  PLT(I,J,K) = PC
    If this decision variable is frozen, i.e., if BS(I,J,K) = 1
    or BT(NT) = 1, go to step 11. If this mine source already
    has treatment specified and pollutant would not be released
    by an inactive instream treatment facility, i.e., if
    D(I,J,K) = 2 and PE = 0.0, go to step 11.

4.  Initialize QMAX to a large number. Examine each node between
    (I,J,K) and the basin outlet to determine QMAX
    using equation C.33.

5.  If natural pollutant would not be released by an inactive
    instream treatment facility; i.e., PE = 0, go to step 6.
    If QMAX $\geq$ PE, go to step 6.
    Activate NT by
    a.  calling subroutine TON,
    b.  setting $x_m$* = 1.
    c.  setting $x_{m+1}$* = 0 if BS(I,J,K) = 0
    d.  calling subroutine ZO.
    Stop, the procedure is complete.

6.  QMAX - PE $\rightarrow$ QMAX
    If $x_m$* represents an instream treatment facility,
    set Z = $x_{m+1}$; otherwise, set Z = $x_m$.
    If Z > 0, go to step 8.
```

```
     PL = P(I,J,K)
     If QMAX $\geq$ PL, go to step 10.
     PL = Pl($\bar{I}$,J,K)
     Call subroutine ZO
     If QMAX $\geq$ PL, go to step 7.
     Z = 2
     PL = 0.0
     Go to step 10.

 7.  Z = 1.
     If RALT(1,I,J,K) > 1, Z = 2
     Go to step 10.

 8.  If Z = 2, go to step 9.
     PL = Pl(I,J,K)
     If QMAX $\geq$ PL, go to step 10.
     Call subroutine ZO
     Z = 2

 9.  PL = 0.0

10.  If $x_m^*$ represents an instream treatment decision,
     $x_m^* = x_m$ and $x_{m+1}^* = Z$; otherwise, $x_m^* = Z$.
     Set PLE = pollutant released from this node based upon
     $x_m^*$ (and $x_{m+1}^*$ if NT > 0).  Adjust downstream values of
     PLT(I,J,K) for this value of PLE.
     This procedure is complete.

11.  Set $x_m^* = x_m$.
     If $x_m^*$ represents an instream treatment decision,
     $x_{m+1}^* = x_{m+1}$.
     This procedure is complete.
```

Determination of the Stopping Vector -

The stopping vector, X_L, is the last vector in the numerical ordering of X that needs to be evaluated in searching for the optimum decision vector. In the cases evaluated by the DWMC model, extending the Lawler-Bell algorithm by efficient selection of a stopping vector permitted reductions in computer effort by several orders of magnitude.

The basic concept behind the calculation of elements of the stopping vector is based upon calculating the maximum admissible pollutant flow at a node; i.e., the maximum possible pollutant flow from all feasible decision vectors. Let POUT be this maximum possible pollutant flow at the next upstream node from the basin outlet, and let PLTMAX be the maximum allowable flow based on the quality standard QS. Consider the following relationship to illustrate the potential economies of a stopping vector. If (I,J,K) are the coordinates of the basin outlet node, then $x_1^L = 0$ if POUT + P(I,J,K) $\leq$ PLTMAX, where x_1^L is the first element of X^L. The above relationship is based upon the observation that the optimal solution would certainly not include expenditures for pollution control if they were not required to meet the quality standard. In addition the above relationship states that no feasible decision vector would merit pollution control expenditures at this node. The potential for making substantial cuts in computer effort is also illustrated by the above relationship since setting x_1^L to zero cuts the number of possible decision vectors into one third of its previous value.

To extend the above illustration, several additional relationships are employed. First, inequalities are needed to specify the situations where $x_1^L = 1$ and $x_1^L = 2$. Also, procedures are used to specify the stopping vector elements at upstream nodes and at instream treatment facilities. In going to upstream nodes, the definition of the upper limit on the allowable pollution flow must change to account for the value of downstream stopping vector elements. For example, setting $x_1^L = 0$ implies that the outlet node will emit pollutant at the rate of P(I,J,K); thus, the maximum allowable pollutant at the next upstream node must be sufficiently low so that an effluent of P(I,J,K) from the last node will be feasible.

The maximum allowable pollutant is, of course, affected by natural pollutant flows. In this procedure for determining the stopping vector, the natural pollutant flows are more readily used in the following form:

PNT(I,J,K) = total natural pollutant flow past node (I,J,K) assuming no instream processors are activated upstream.

In fact all pollution flows used in this procedure are computed relative to the natural pollution flow of PNT(I,J,K) as an origin. That is, the absolute flows are POUT + PNT(I,J,K) and PLTMAX + PNT(I,J,K).

The stopping vector X_L is actually recorded by program ALCOT using the following array elements:

DL(I,J,K) = value of the stopping decision vector at mine source (I,J,K)

DINTL(NT) = value of the stopping decision vector at instream treatment site NT.

The relationships presented below are employed by program ALCOT in determining elements of the arrays DL and DINTL. These relationships are used to determine the stopping vector element at the node (I,J,K);

thus, (I+1,J,K) is the next downstream node and its stopping vector element or elements have already been determined. Let PLTMAX be the maximum allowable pollutant flow that was used in determining the stopping vector elements at node (I+1,J,K). Then, PLTMAX at node (I,J,K) is determined by

$$\text{Min}\left[\text{PLTMAX} - \text{PP}, \frac{\text{QS} \cdot 10^{-6}}{1 - \text{QS} \cdot 10^{-6}} \cdot \text{Q}(\text{I,J,K}) - \text{PNT}(\text{I,J,K})\right] \rightarrow \text{PLTMAX}, \tag{C.34}$$

where PP = pollutant produced at node (I+1,J,K) based upon stopping vector decisions at that node.

Note that the pollutant flows are regarded as being relative to the natural pollutant flow which would occur if no upstream instream processors were employed. Subroutine PTMX takes an allowable pollutant flow at a node, such as PLTMAX, and computes the maximum flow less than or equal to the allowable flow which would occur. In the procedure, PTMX computes POUT, the maximum admissible flow at node (I-1,J,K) that is less than or equal to PLTMAX. If (I,J,K) is the head of a stream, POUT is regarded as zero. Assuming that (I,J,K) is not a potential instream processor site, then

$$\begin{aligned}
\text{DL}(\text{I,J,K}) &= 0 \text{ if POUT} + \text{P}(\text{I,J,K}) \leq \text{PLTMAX}; \\
\text{DL}(\text{I,J,K}) &= 1 \text{ if POUT} + \text{P1}(\text{I,J,K}) \leq \text{PLTMAX} < \text{POUT} + \text{P}(\text{I,J,K}) \text{ and} \\
&\quad \text{RALT}(1,\text{I,J,K}) < 2, \text{ and} \\
\text{DL}(\text{I,J,K}) &= 2 \text{ if otherwise.}
\end{aligned} \tag{C.35}$$

If (I,J,K) is a possible instream processor site, then a check must be made to determine whether the maximum pollutant flow will merit the consideration of an instream processor. To perform this check, compute POUT without restrictions from upstream decisions; i.e., without considering PLTMAX as computed by equation C.34; then the only restriction on POUT is that it is an admissible flow at node (I-1,J,K). Assuming that mine source treatment is always cheaper than instream treatment, the only case in which the maximum flow will merit activating the instream processor is when POUT > PLTMAX, where PLTMAX is determined by C.34. This is true because in any other case a feasible solution can be obtained by mine source treatment.

In order to initiate the above procedure at the outlet of a stream, some special rules need to be invoked to obtain an initial value of PLTMAX, and they are discussed below. If the stream is the basin outlet stream; i.e., the level 3 stream, then PLTMAX can be based purely on the quality standard. That is,

$$\text{PLTMAX} = \frac{\text{QS} \cdot 10^{-6}}{1 - \text{QS} \cdot 10^{6}} \cdot \text{Q(I,J,K)} - \text{PNT(I,J,K)}, \qquad \text{(C.36)}$$

where (I,J,K) is the basin outlet node. PLTMAX is not based purely on the quality standard if the stream is a level 2 or a level 1 stream. Designate this stream as stream (J,K) where K equals 1 or 2. Then, PLTMAX is determined from calculations made by subroutine PTMX to determine the maximum allowable flow POUT at the confluence node receiving flow from stream (J,K). As requried, PTMX records the flow to a confluence node in the variables POD, the input from the tributary, and PJD, the input to the confluence node from upstream; thus,

$$\text{POUT} = \text{POD} + \text{PJD}.$$

Then, program ALCOT sets PLTMAX equal to PJD for calculations upstream to a confluence node, and ALCOT stores POD for later use by setting PLT(J,K) = POD.

The reader can refer to the flow chart of program ALCOT, Figure C.14, for a detailed presentation of the computational procedure for determining values of the DL and DINTL arrays given maximum admissible flows. The method for determining the maximum admissible flows is presented in the following discussion.

Subroutine PTMX is given an upper limit, PLTMAX, on pollutant flow past a particular node, (IM,JM,KM), and the subroutine is to calculate the maximum pollutant flow, POUT, which meets quality standards and is less than PLTMAX. PTMX uses a recursive procedure to determine POUT starting from the head of level one streams and working downstream to (I,J,K). The procedure employed treats confluence nodes in a completely different manner than it treats nodes void of stream confluences. The procedure for nodes not having stream confluences is described first, and then the method is extended to account for stream confluences.

The basic method for determining POUT at nodes without stream confluences involves calculating a maximum pollutant flow, called PMAX, and using the value of PMAX to determine the corresponding value of PMAX at the next downstream node. The following relationships are used to specify PMAX at node (I,J,K) given the value of PMAX at node (I-1,J,K). Let

$$\text{QST} = \text{Min}\left[\text{PU}, \frac{\text{QS} \cdot 10^{-6}}{1 - \text{QS} \cdot 10^{-6}} \cdot \text{Q(I,J,K)} - \text{PNT(I,J,K)}\right],$$

where PU is initially set equal to PLTMAX. QST represents the upper limit on pollution flow at node (I,J,K). Also, let

$$\text{QSN} = \text{QST} + \text{FNT}(I,J,K) - \text{PN}(I,J,K),$$

where QSN is the upper limit on pollution flow which can be emitted from node (I,J,K) when maximum control actions are exerted upstream. Recall that PN(I,J,K) is the natural pollution flow under the assumption that all upstream instream processors are activated.

$$\begin{aligned}&\text{If } P(I,J,K) > \text{QSN}, \text{ then PMAXO} = \text{PMAX}\\ &\text{If } P(I,J,K) + \text{PMAX} \leq \text{QST}, \text{ then PMAXO} = P(I,J,K) + \text{PMAX}\end{aligned} \qquad (C.37)$$

$$\begin{aligned}&\text{If } P1(I,J,K) > \text{QSN}, \text{ then PMAXO} \rightarrow \text{PMAXO}\\ &\text{If } P1(I,J,K) + \text{PMAX} \leq \text{QST}, \text{ then Max}\Big[\text{PMAXO}, P1(I,J,K) + \text{PMAX}\Big] \rightarrow \text{PMAXO}\end{aligned} \qquad (C.38)$$

after evaluating the above expressions,

$$\text{Max}(\text{PMAX},\text{PMAXO}) \rightarrow \text{PMAX}\,. \qquad (C.39)$$

The above relationships, if satisfied, clearly lead to a new value of PMAX and they must be evaluated in the order shown. However, the result is unclear if one or more of the three conditions listed below exist:

$$\begin{aligned}&\phantom{\text{and}}\quad 1.\ P(I,J,K) \leq \text{QSN}\\ &\text{and}\quad\ \ P(I,J,K) + \text{PMAX} > \text{QST},\end{aligned} \qquad (C.40)$$

$$\begin{aligned}&\text{or}\quad\ \ 2.\ P1(I,J,K) \leq \text{QSN}\\ &\text{and}\quad\ \ P1(I,J,K) + \text{PMAX} > \text{QST},\end{aligned} \qquad (C.41)$$

$$\text{or}\quad 3.\ \text{PMAX} > \text{QST} \qquad (C.42)$$

In other words, there may exist a pollution flow less than PMAX at the next upstream node which could result in a larger maximum flow rate at (I,J,K). The three cases depicted by C.40, C.41, and C.42 are called uncertain maxima until the existence of pollution flows less than PMAX is known.

Once an uncertain maximum is encountered, the procedure is halted temporarily until the uncertain maximum can be clarified. Clarification of the uncertain maximum is performed by setting PU in the equation for QST to a new value and restarting the procedure at the stream head. The new value of PU is

$$PU = QST - P(I,J,K) \tag{C.43}$$

if equation C.40 generated the uncertain maximum or

$$PU = QST - P1(I,J,K) \tag{C.44}$$

if equation C.41 generated the uncertain maximum or

$$PU = QST \tag{C.45}$$

if otherwise. When the uncertain maximum is first encountered, subroutine PTMX calls subroutine PTMAX in an attempt to resolve the uncertain maximum. Assuming that PTMAX can determine the maximum flow less than PU without creating a new uncertain maximum, then PTMAX sets PTM1 equal to this maximum flow. The uncertain maximum is resolved by one of the three relationships below.

1. If equation C.40 generated the uncertain maximum, then

   $$PMAXO = Max[PTM1 + P(I,J,K), PMAX] \tag{C.46}$$

   and the procedure is restarted at equation C.38.

2. If equation C.41 generated the uncertain maximum, then

   $$Max[PMAXO, PTM1 + P(I,J,K)] \rightarrow PMAXO \tag{C.47}$$

   and the procedure is restarted at equation C.39.

3. If equation C.42 generated the uncertain maximum, then

   $$Max(PMAXO, PTM1) \rightarrow PMAX \tag{C.48}$$

The procedure described above may generate a number of uncertain maxima, and information concerning these uncertain maxima is recorded on an uncertain maximum list. That is, in the event PTMAX encounters a new uncertain maximum, an entry on the uncertain maximum list is created. This entry records the situation as it existed when the original uncertain maximum was created. The values stored are:

[MUI(L), MUJ(L), MUK(L)] = objective coordinates for entry L on the uncertain maximum list.

PUL(L) = upper limit on pollution flow rate to be used in formula for QST for entry L on the uncertain maximum list.

PMAL(L) = maximum pollution flow rate for entry L on the uncertain maximum list.

PMALO(L) = maximum pollution flow rate corresponding to PMAXO for entry L on the uncertain maximum list.

MU = number of entries on the uncertain maximum list.

The objective node for each entry is the node to which the procedure was directed when the Lth uncertain maximum occurred; thus, [MUI(1), MUJ(1), MUK(1)] is always the node for which PTMX is computing the maximum pollution flow less than or equal to PLTMAX. That is, MUI(1) = IM, MUJ(1) = JM, and MUK(1) = KM. Note that [MUI(L+1), MUJ(L+1), MUK(L+1)] is the node at which the Lth uncertain maximum occurred. After storing an entry on the uncertain maximum list, PTMX then calls PTMAX again in an attempt to resolve the newest uncertain maximum. Once PTMAX is successful in resolving an uncertain maximum by equation C.46, C.47, or C.48, PTMX then restarts its basic procedure in an attempt to resolve the last entry on the uncertain maximum list using equations C.37, C.38, and C.39.

In the event (I,J,K) is also a potential instream processor, then checks are made to determine whether activating the instream processor will increase the value of PMAX. In any event, PMAX must be less than or equal to PU, the upper limit on pollution flow rate. Recall that PMAX and PU are relative to an origin at PNT(I,J,K). Thus, the output of an instream processor would be -PNT(I,J,K) regardless of the input so long as the input is positive. The assumption is made that proper selection of upstream decision variables can always be made to yield a positive input. The following relationship is used at an instream processor to determine a new value for PMAX.

$$\text{If PMAX} < -\text{PNT(I,J,K) and QST} \geq -\text{PNT(I,J,K), then} -\text{PNT(I,J,K)} \rightarrow \text{PMAX} \qquad (\text{C.49})$$

Calculation of PMAX values downstream of a confluence node is a more complex process. The confluence node is characterized by the fact that there may be a large number of possible pollution flowrates from both the tributary and the main stream. Accordingly, subroutine PTMX builds a list of the possible flow rates that are input to a confluence node. Each confluence node is designated in the list by the tributary stream feeding the confluence node, and each tributary stream is either a level 1 or a level 2 stream. The variables used to tabulate the list of possible inputs to confluence nodes are defined below.

PO1DIS(I,J) = Ith admissible output pollutant flow from level 1 stream J.

PO2DIS(I,J) = Ith admissible output pollutant flow from level 2 stream J.

PJ1DIS(I,J) = Ith admissible pollutant flow from the main stream and input to the confluence node receiving flow from level 1 stream J.

PJ2DIS(I,J) = Ith admissible pollutant flow from the main stream and input to the confluence node receiving flow from level 2 stream J.

Each of the above arrays is ordered so that the Ith admissible pollutant flow is greater than the I+1st admissible flow. In understanding the procedure, it is important to note that the first value computed as output from a stream and as input from the main stream to a confluence node will always be the most unconstrained value or largest value. The computational procedure already provides for calculating values of PO1DIS(1,J) PO2DIS(1,J), PJ1DIS(1,J), and PJ2DIS(1,J). Smaller flow values, or values for values of the subscript I greater than 1, have to be computed by special request as needed.

Subroutine CONO is used by PTMX to determine the output value of PMAX from a confluence node. If insufficient values for the node input distribution values are available, then CONO generates an entry to the uncertain maximum list and restarts the computational procedure for PTMX with a value of PU slightly lower than the last tabulated value of an input distribution array; i.e., PO1DIS, PO2DIS, PJ1DIS, or PJ2DIS. Thus, PTMX will compute the next lower input distribution value. Assuming that the confluence node (I,J,2) in question receives flow from level 1 stream NF, then the procedure used by CONO to determine if sufficient node input distribution values have been calculated is summarized below:

1. Set QST = upper limit on pollution flow from confluence node (I,J,2)
   QSN = QST - PN(I,J,2) + PNT(I,J,2)

CHK = lowest possible input from stream J to the confluence node
CHK = QST - QSN
CHO = lowest possible output from stream NF
CHO = PN[ND(NF,1), NF, 1] - PNT[ND(NF,1), NF, 1]
PP = P(I,J,2)
If PP $\leq$ QSN, go to step 2
PP = P1(I,J,2)
If PP $\leq$ QSN, to to step 2
PP = 0.0.

2. Set NO = number of values tabulated in the array PO1DIS for stream NF
   NU = number of values tabulated in the array PJ1DIS for stream NF
   PCK = QST - PP
   If PO1DIS(NO,NF) $\leq$ CHO, go to step 4
   If PO1DIS(NO,NF) + PJ1DIS(1,NF) $\leq$ PCK, go to step 4.

3. Set PO1DIS(NO,NF) - EPSN → PU, where EPSN is a small number
   Go to step 6.

4. If PJ1DIS(NU,NF) $\leq$ CHK, go to step 7
   If PJ1DIS(NU,NF) + PO1DIS(1,NF) $\leq$ PCK, go to step 7.

5. Set PJ1DIS(NU,NF) - EPSN → PU, where EPSN is a small number.

6. Return to subroutine PTMX.
   Record current situation in uncertain maximum list.
   Restart procedure to calculate another value for the PO1DIS or PJ1DIS array.
   Procedure is complete.

7. Sufficient values have been computed for the PO1DIS and PJ1DIS arrays.
   Compute maximum output from this confluence node.

The basic elements of the procedure to compute the stopping vector X_L include the above method for determining the maximum output from a confluence node and the maximum output from nodes not having stream confluences. When the procedure finally reaches the node (IM,JM,KM), then PTMX has completed the calculation of the maximum flow at this node; thus, POUT is set equal to PMAX, and the computations are complete. Then, program ALCOT uses this value of POUT in determining the elements of the stopping vector or the DL and DINTL arrays. See the flowcharts of program ALCOT and subroutines PTMX, PTMAX, and CONO in Figures C.14, C.20, C.19, and C.15 for more details concerning this procedure.

Overall Procedure of the DWMC Algorithm -

The basic computational procedure followed by program ALCOT in determining the least cost solution for satisfying the quality constraint of QSppm is described in this section. The optimal set of decisions is recorded in the two arrays whose elements are defined below:

O(I,J,K) = optimal value of D(I,J,K)
OINT(NT) = optimal value of DINT(NT)

To identify when a better solution is found, the least cost value of the currently recorded optimal solution is recorded in the variable TC. As new feasible decision vectors are found with less cost than TC, the value of TC is updated, and the decision vector is stored in the O and OINT arrays.

The procedure followed by ALCOT in determining an optimal solution is summarized below.

1. Initialize variables
   TC = 10^{31}
   Set CIST(NT) = 10^{31} and OI(NT) = 0 for NT = 1,2,...,NINST.
   Set D and DINT arrays to zero values.
   Compute the elements of the stopping vector; i.e., the elements of the DL and DINTL arrays.

2. Call subroutine NEXFES to determine the next feasible solution which is recorded in the D and DINT arrays.

3. Call function TCOST to compute the total cost for the decision vector given by the D and DINT arrays. Record the result in the variable TTC.

4. If TC < TTC, go to step 5.
   Record a new optimal solution in the O and OINT arrays.
   Set TC = TTC.

5. Set NT = 1.

6. If OI(NT) ≠ 0 or DINT(NT) ≠ 1, go to step 7.
   If CIST(NT) < TTC, go to step 7.
   Record a new optimal upstream solution for instream processor NT in the IO and IOI arrays.
   Set CIST(NT) = TTC.

7. If NT ≥ NINST, go to step 8;
   otherwise, NT + 1 → NT, and go to step 6.

8. Using the procedure outlined in extension 2, skip to the next decision vector in the numerical ordering which could have less cost than TC. Record the decision vector in the D and DINT arrays.

9. Check to determine whether the decision vector given by the D and DINT arrays is past the vector given by the DL and DINTL arrays in the numerical ordering. If so, go to step 11.

10. Go to step 2.

11. The optimal solution is recorded in the O and OINT arrays, and its cost is TC.

PROGRAM MAXEF

Purpose

Program MAXEF determines the maximum effectiveness resource allocation to control mine drainage pollution within a watershed for a specified budget constraint.

Method

A stream network is defined and decision nodes indicated. At each mine source decisions can be made to treat or not to treat, to abate or not to abate; and all possible combinations of these decisions are considered. At each potential instream processor site, the site may be implemented or not used. The cost and effectiveness of each decision at each node is determined. For a given maximum budget allocation, the most effective feasible pollution control scheme for the network is then determined using a modification of the Lawler-Bell alogrithm. Effectiveness is calculated as a function of the maximum pollutant concentration along each reach between decision nodes.

Definition of Variables

AP(I,J,K) = Annual pollutant load emitted from source I on stream J of level K (kg).

AP1(I,J,K) = Annual pollutant load emitted from source I on stream J of level K when source allocation alternative 1 is selected (kg).

APA(I,J,K) = Annual pollutant loading emitted from source I on stream J of level K after abatement (kg).

APN(I,J,K) = Annual pollutant load at node I on stream J of level K due to natural sources (kg).

APINS(NT) = Annual pollutant flow at potential instream treatment site NT under the assumption all upstream sources have no pollution control measures (kg). Exception is made at those sources where cost to abate is less than variable cost of treatment.

APST(NT) = Annual pollutant flow just upstream of potential treatment site NT (kg).

BS(I,J,K) = { -1 if abatement or treatment must be performed at site (I,J,K),
0 otherwise.

BUD = Maximum allowable resource cost.

BV(J) = Basic value for the Jth maximum pollution concentration interval.

C(1,I,J,K) = Cost to abate source I on stream J of level K.

C(2,I,J,K) = Fixed cost to treat at source I on stream J of level K.

CI(NT) = Fixed cost to perform instream treatment at instream treatment site NT.

CALT(ID,I,J,K) = Cost of resource allocation alternative ID for source (I,J,K).

D(I,J,K) = Allocation alternative selected for source (I,J,K).

D(I,J,K) = { 2 if no mine drainage control measures are to be performed at source (I,J,K).
1 if cheapest control measure is to be performed at source (I,J,K).
0 if treatment is to be performed at source (I,J,K).

DINT(NT) = { 0 if instream treatment is to be performed at instream treatment site number NT,
1 if otherwise.

DINTL(NT) = Value of DINT(NT) in stopping vector.

DL(I,J,K) = Value of D(I,J,K) in stopping vector.

DOUT = { True if next solution vector is to be written out.
False if otherwise.

EF = Optimal value of pollution control effectiveness.

EFF = Trial value of pollution control effectiveness.

INT(I,J,K) = { NT if node (I,J,K) is a treatment node where NT is the treatment site number.
0 if otherwise.

JN(I,J,K) = {0 if node I on level K+1 stream J is not a confluence node.
NF otherwise where NF is the stream of level K feeding node I on level K+1 stream J.

KBW = KU - KL + 1.

KL = Level of the lowest level stream represented.

KLA = KL + 1.

KNINT(I,J,K) = {1 if node (I,J,K) feeds an active instream treatment site.
0, if otherwise.

KNT = Number of times the criterion function has been evaluated.

KNTLIM = Upper limit on the value of KNT for this run.

KOPT = Value of KNT when the optimal solution was evaluated.

KOUT = The interval between output of solution vectors.

KSINT(NT) = {1 if instream treatment site NT feeds an active instream site, and
0 if otherwise.

KU = Level of the highest level stream represented.

MNINST = Dimensioned value of all arrays subscripted by instream treatment site number.

MNO = Dimensioned value of the node number subscript in all arrays subscripted by node number.

MNOS = MNO•MNS.

MNS = Dimensioned value of the stream number subscript in all arrays subscripted by node number.

MS = {0 if neither abatement nor treatment is to be performed.
1 if abatement but no source treatment is to be performed.
2 if source treatment but no abatement is to be performed.
3 if both abatement and source treatment is to be performed.

ND(J,K) = Total number of nodes on stream J of level K.

NFN(J,K) = Confluence node on level K+1 stream receiving flow from level K stream J.

NFS(J) = Level 2 stream receiving flow from level 1 stream J.

NI = Total number of pollution concentration intervals.

NINST = Total number of possible instream treatment site locations.

NS(K) = Total number of streams of level K.

NSO = 3•MNOS.

O(I,J,K) = Optimal allocation alternative selected for source (I,J,K).

OINT(NT) = Optimal value of DINT(NT).

P(I,J,K) = Pollutant loading emitted from source I on stream J of level K (kg/hr).

P1(I,J,K) = Pollutant output for resource allocation alternative 1 for source (I,J,K) (kg/hr).

PA(I,J,K) = Pollutant loading emitted from source I on stream J of level K after abatement (kg/hr).

PL(I) = Level of Ith stream to process.

PN(I,J,K) = Natural pollutant incremental flow occurring at node (I,J,K) (kg/hr).

PS(I) = Ith stream to process.

PT(J,K) = Pollutant input from stream J of level K, K = 1,2.

Q(I,J,K) = Stream flow at node (I,J,K) excluding the pollutant. (Input as cubic meters per second converted to (kg/hr)).

QJ(J) = Upper limit on the maximum pollution concentration for the Jth interval (ppm as input-converted to decimal fraction).

R(I,J,K) = Relative importance of the stream reach between nodes (I+1,J,K) and (I,J,K).

RALT(ID,I,J,K) = Value of MS for source (I,J,K) for resource allocation alternative ID (ID = 1 for lowest cost alternative, ID = 2 for alternative involving source treatment).

TC = Cost of maximum effectiveness solution.

TTC = Trial total cost value.

VC = Annual variable cost to treat one unit of pollution ($/kg).

Input Data:

Card Number	Variable Name	Columns Used	Format
1	NS(1)	1-5	Integer
1	NS(2)	6-10	Integer
1	NS(3)	11-15	Integer
1	VC	16-25	Real
1	MNO	26-30	Integer
1	MNS	31-35	Integer
1	Note: Columns 36-45 are blank		
1	NINST	46-50	Integer
1	MNIST	51-55	Integer
1	KOUT	56-60	Integer
2	NI	1-5	Integer
2	BUD	6-15	Real
3	BV(1)	1-10	Real
3	QJ(1)	11-20	Real
	.		
	.		
	.		
NI+2	BV(NI)	1-10	Real
NI+2	QJ(NI)	11-20	Real
NI+3 (see note 1)	ND(1,1)	1-5	Integer
	ND(2,1)	6-10	Integer
	ND(3,1)	11-15	Integer
	.		
	.		Integer
	.		
	ND(NS(1),1)	(5NS(1)-4)—5NS(1)	Integer
NI+5-KL	ND(1,2)	1-5	Integer
	.		
	.		
	.		
NI+6-KL (see note 2)	ND(1,3)	1-5	Integer
	.		
	.		
	.		
	ND(NS(3),3)	(5NS(3)-4)—5NS(3)	Integer
NI+7-KL	JN(1,1,1)	1-5	Integer
	JN(2,1,1)	6-10	Integer
	JN(3,1,1)	11-15	Integer
	.		
	.		
	.		
	JN(ND(1,2),1,1)	(5ND(1,2)-4)—5ND(1,2)	Integer
NI+8-KL	JN(1,2,1)	1-5	Integer
	JN(2,2,1)	6-10	Integer

	. . .		
(see note 3)	JN(ND(1,3),1,2)	(5ND(1,3)-4)—5ND(1,3)	Integer
NI+8+NS(2)-KL	P(1,1,1)	1-10	Real
	PA(1,1,1)	11-20	Real
	AP(1,1,1)	21-30	Real
	APA(1,1,1)	31-40	Real
	Q(1,1,1)	41-50	Real
	PN(1,1,1)	51-60	Real
	APN(1,1,1)	61-70	Real
	INT(1,1,1)	71-75	Integer
NI+9+NS(2)-KL	C(1,1,1,1)	1-10	Real
	C(2,1,1,1)	11-20	Real
	R(1,1,1)	21-30	Real
NI+10+NS(2)-KL	P(2,1,1)	1-10	Real
	PA(2,1,1)	11-20	Real
	AP(2,1,1)	21-30	Real
	APA(2,1,1)	31-40	Real
	Q(2,1,1)	41-50	Real
	PN(2,1,1)	51-60	Real
	APN(2,1,1)	61-70	Real
	INT(2,1,1)	71-75	Integer
NI+11+NS(2)-KL	C(1,2,1,1)	1-10	Real
	C(2,2,1,1)	11-20	Real
(see note 4)	R(2,1,1)	21-30	Real
NI+7+NS(2)-KL +2W	CI(1)	1-10	Real
	CI(2)	11-20	Real
	CI(3)	21-30	Real
	. .		
	CI(8)	71-80	Real
NI+7+NS(2)-KL +2W+NIN	CI(9)	1-10	Real
	. .		
(see note 5)	CI(NINST)	(10 NINST-80 NIN)-(10 NINST-80 NIN)	Real
NI+8+NS(2)-KL +2W+NIN	KOPT	1-10	Integer
(see note 6)	KNT	11-20	Integer
	KNTLIM	21-30	Integer
NI+9+NS(2)-KL +2W+NIN	EF		Integer
(see note 7)	TC		Integer
	D		Integer
	BS		Integer
	O		Integer
	KNINT		Integer
	DINT		Integer
	KSINT		Integer
	OINT		Integer

Notes:

1. The number of the first ND array card depends upon the value of NI

2. The number of ND array cards is variable, depending on the number of stream levels. For a 3-level network there are three ND array cards; for a 2-level network there are two ND array cards; for a 1-level network there is one ND array card. These input data instructions assume a 3-level network.

3. The number of JN array cards depends upon the total number of streams of all levels. JN array cards are sequenced numerically according to the lowest level streams, next lowest level, etc. Note that if there is only a level-three stream in the network, this sequence of cards is skipped. If there are only level-two streams and a level-three stream, there is only one JN array card. If there are level-one streams, a JN array card is provided for each level-two and level-three stream. The above input data instructions assume three stream levels exist.

4. There are two cards entered for each node using the format specified for the following inputs: P(I,J,K), PA(I,J,K), AP(I,J,K), APA(I,J,K), Q(I,J,K), PN(I,J,K), APN(I,J,K), INT(I,J,K), C(1,I,J,K), C(2,I,J,K), and R(I,J,K). The nodes are entered in sequence starting with the first node on the first stream of level KL. Nodes for this stream are entered, the next stream of level KL is entered, one node at a time. After recording the data for all level KL streams, then level KL + 1 streams are entered. The last node entered is the last node on the level 3 stream. A total of

   $$W = \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} ND(J,K) \text{ nodes are entered on 2W cards.}$$

5. $$NIN = \begin{cases} 1 \text{ if } NINST > 8 \\ 0 \text{ if otherwise} \end{cases}$$

6. For the initial run of a basin to be analyzed, set KOPT and KNT to zero.

7. This sequence of cards is not used on an initial run and is only used when a run restarts after KNT solution vectors have been evaluated. If an optimal solution is not realized within KNTLIM iterations, a sequence of cards will be punched. The first card punched contains current values of KOPT and KNT, to which a value of KNTLIM must be added for subsequent runs. The remaining cards are placed behind the one containing the new values of KOPT, KNT, KNTLIM to restart the run. The previous KOPT, KNT, KNTLIM card

is removed and the program may be restarted using the new data. The new data give values for the D, BS, O, KNINT, DINT, KSINT, OINT arrays as shown.

Common Areas and Contents

COMMON/KST/KNINT, KSINT, APINS, BUD, APST

COMMON/ZER/MNS

COMMON/EFF/BV, QJ, NI

COMMON/NEX/MNOS, MND, NS, ND, D, INT, DINT, BS, R, NFN, NFS, PN, P1,
  Q, P, KL, KU, KBW, KLA

COMMON/TCO/JN, PT, AP, AP1, CALT, CI, VC, APN

COMMON/TONN/NSO, RALT

Subroutines Required

Function CABAT - calculates cost to perform abatement at all nodes upstream of (I,J,K) where abatement is cheaper than treatment variable cost.

Function CSAV - determines the savings in annual pollutant load at a downstream instream processor if decision ID is implemented at source (I,J,K). If $ID < 0$, then the decision is to implement a new instream processor is activated at (I,J,K).

Function EFFECT - determines the effectiveness of the solution vector given by the D and DINT arrays.

Subroutine ERROR - used to abort the run, operate a traceback in the sequence of routines called, and force a core dump when an error is detected in the program.

Subroutine NEFESE - used to determine the next feasible solution. The current solution provided by the DINT and D arrays is obtained if it is feasible.

Subroutine TOFE - processes upstream decision and status variables when an instream treatment processor is deleted.

Subroutine TONE - processes upstream decision and status variables when an instream treatment processor is implemented.

Subroutine ZOE - zeroes out all decision variables that are lower order than (I,J,K).

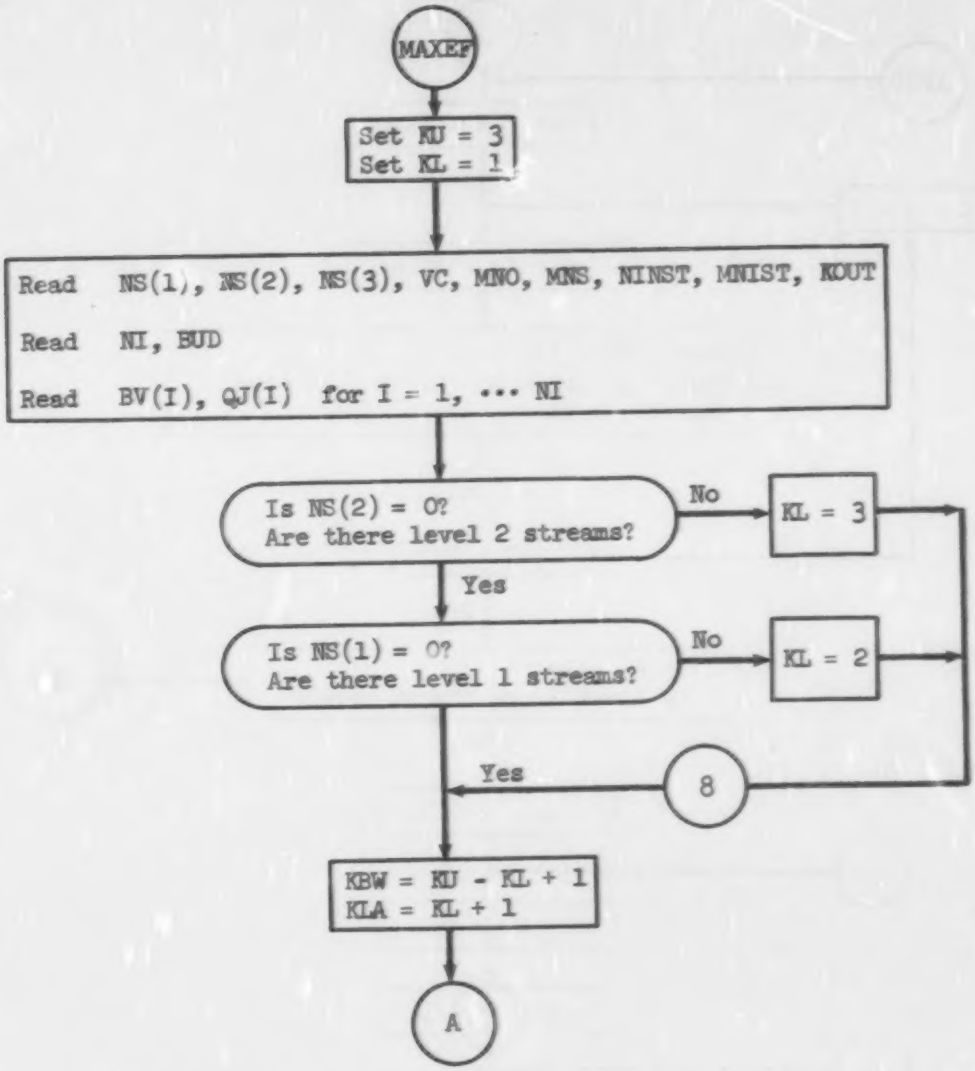

MAXEF
Set KU = 3
Set KL = 1
Read NS(1), NS(2), NS(3), VC, MNO, MNS, NINST, MNIST, KOUT
Read NI, BUD
Read BV(I), QJ(I) for I = 1, ··· NI
Is NS(2) = 0?
Are there level 2 streams?
No
KL = 3
Yes
Is NS(1) = 0?
Are there level 1 streams?
No
KL = 2
Yes
8
KBW = KU - KL + 1
KLA = KL + 1
A

Figure C.5 Program MAXEF

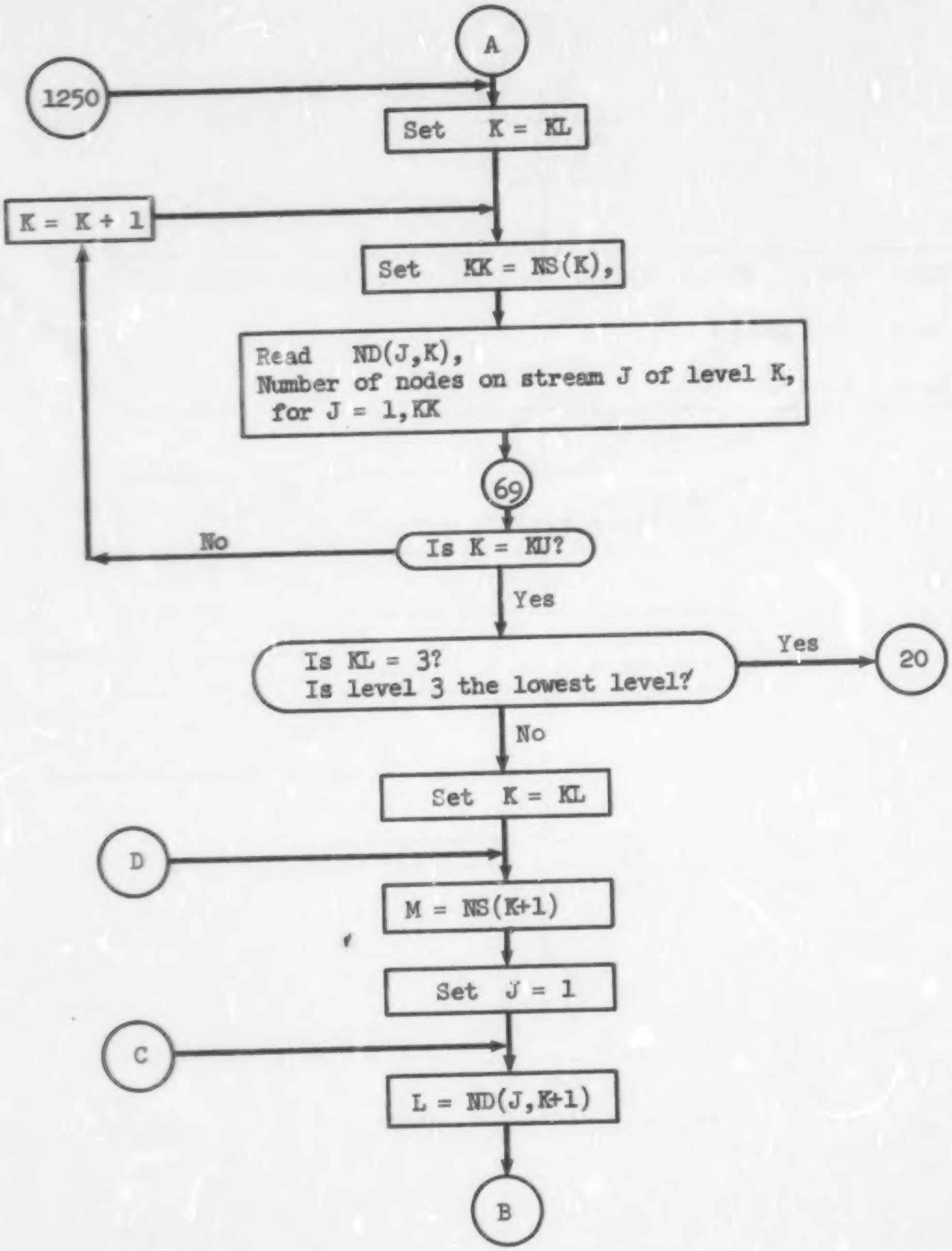

A
1250
Set K = KL
K = K + 1
Set KK = NS(K),
Read ND(J,K),
Number of nodes on stream J of level K,
for J = 1,KK
69
No
Is K = KU?
Yes
Is KL = 3?
Is level 3 the lowest level?
Yes
20
No
Set K = KL
D
M = NS(K+1)
Set J = 1
C
L = ND(J,K+1)
B

Figure C.5 Program MAXEF

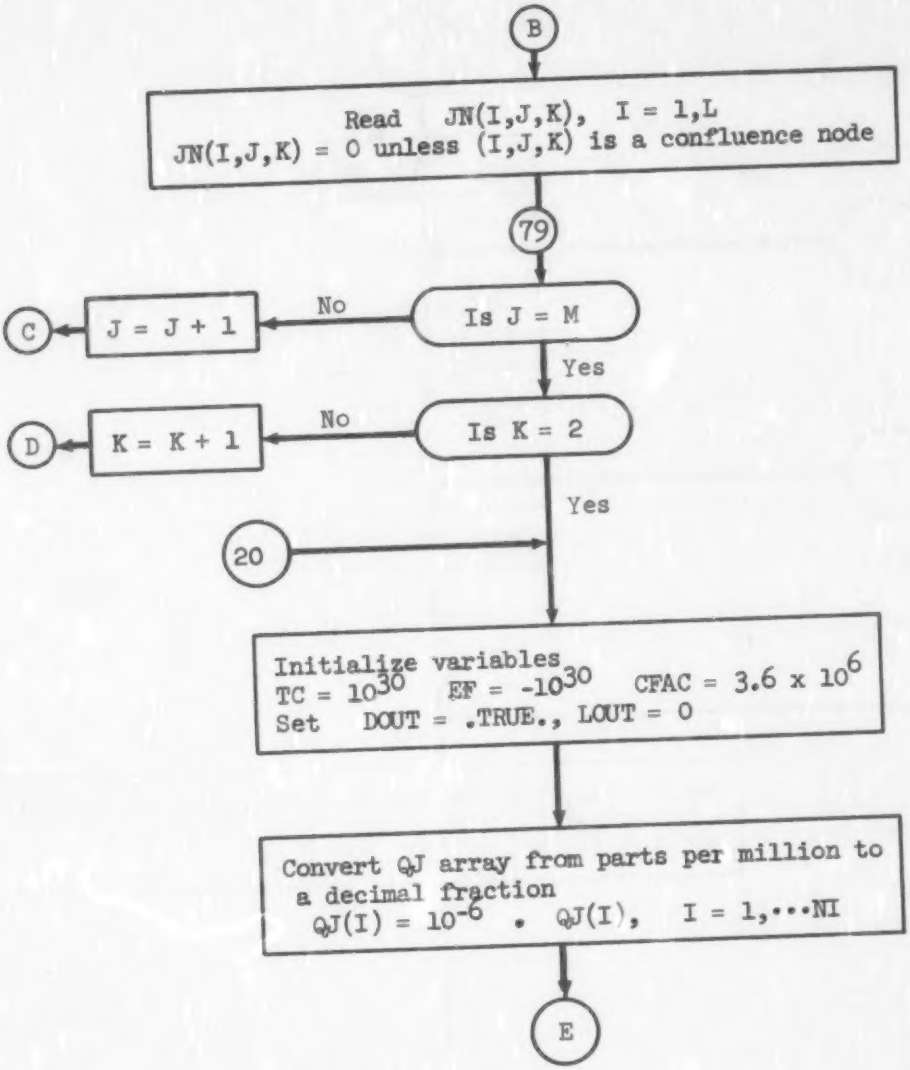

B
Read JN(I,J,K), I = 1,L
JN(I,J,K) = 0 unless (I,J,K) is a confluence node
79
Is J = M
No
J = J + 1
C
Yes
Is K = 2
No
K = K + 1
D
Yes
20
Initialize variables
TC = 10^30 EF = -10^30 CFAC = 3.6 x 10^6
Set DOUT = .TRUE., LOUT = 0
Convert QJ array from parts per million to a decimal fraction
QJ(I) = 10^-6 • QJ(I), I = 1,•••NI
E

Figure C.5 Program MAXEF

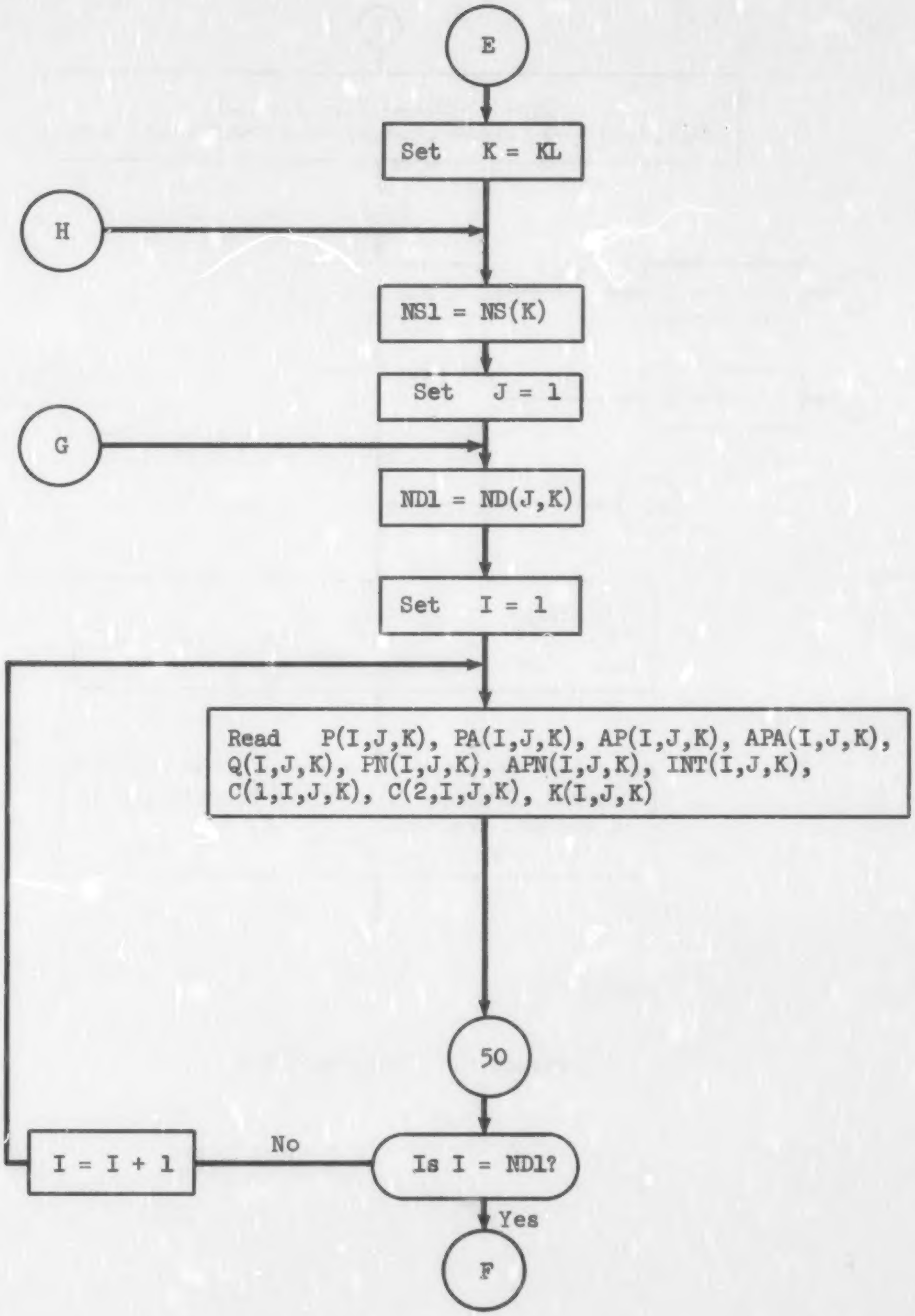

E
Set K = KL
H
NS1 = NS(K)
Set J = 1
G
ND1 = ND(J,K)
Set I = 1
Read P(I,J,K), PA(I,J,K), AP(I,J,K), APA(I,J,K), Q(I,J,K), PN(I,J,K), APN(I,J,K), INT(I,J,K), C(1,I,J,K), C(2,I,J,K), K(I,J,K)
50
I = I + 1
No
Is I = ND1?
Yes
F

Figure C.5 Program MAXEF

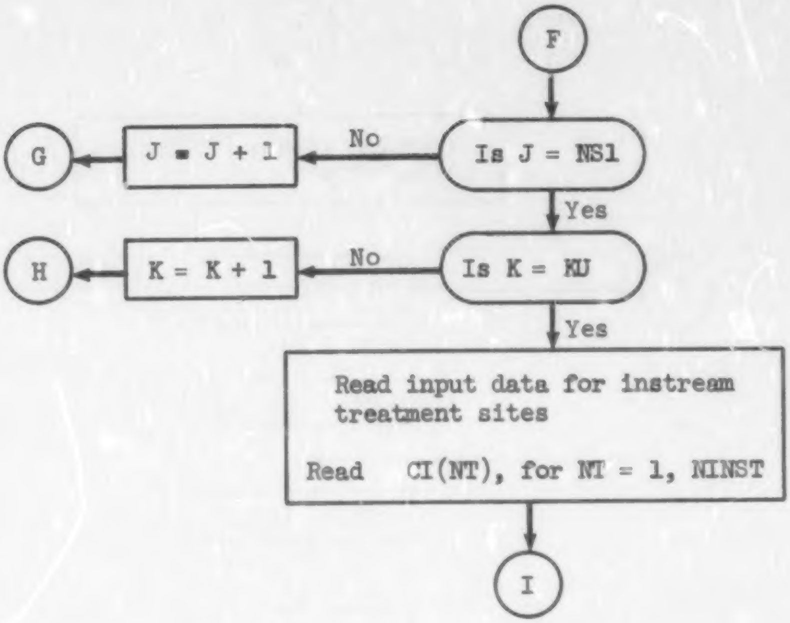

F
Is J = NS1
No
J = J + 1
G
Yes
Is K = KU
No
K = K + 1
H
Yes
Read input data for instream treatment sites
Read CI(NT), for NT = 1, NINST
I

Figure C.5 Program MAXEF

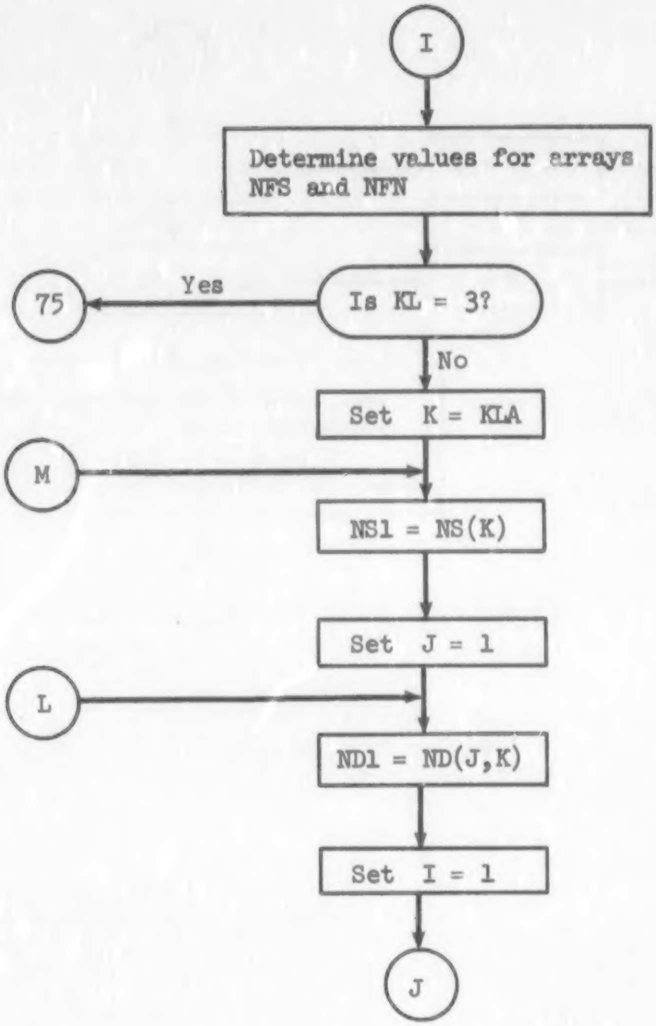

I
Determine values for arrays NFS and NFN
Is KL = 3?
Yes
75
No
Set K = KLA
M
NS1 = NS(K)
Set J = 1
L
ND1 = ND(J,K)
Set I = 1
J

Figure C.5 Program MAXEF

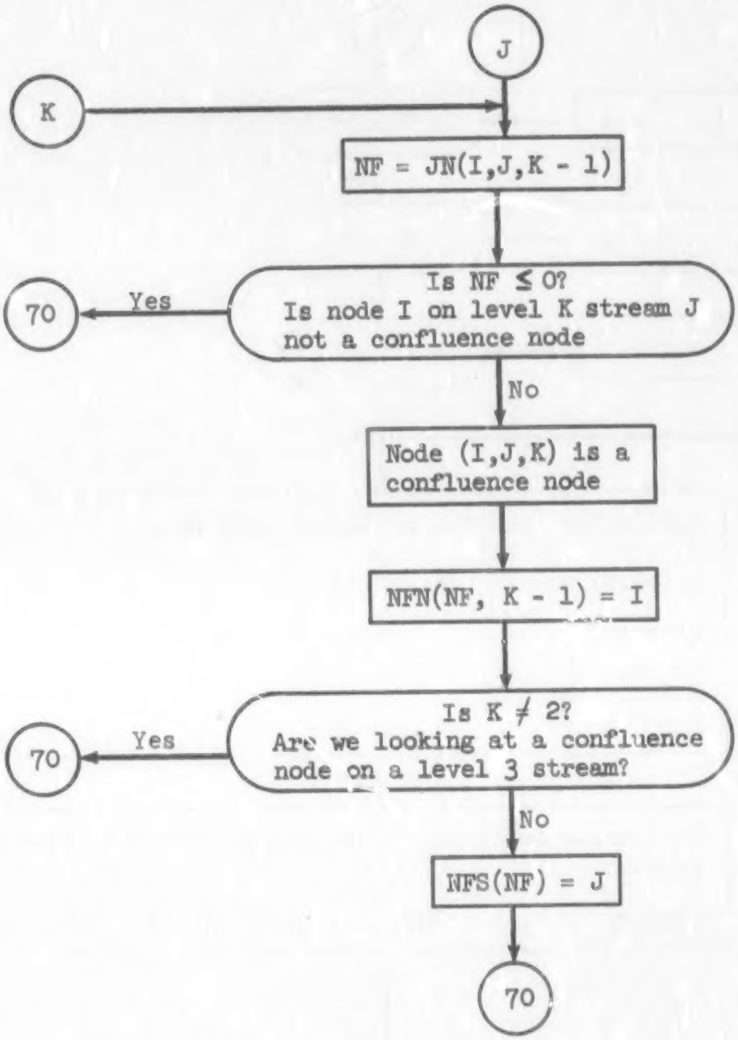

J
K
NF = JN(I,J,K - 1)
Is NF ≤ 0?
Is node I on level K stream J not a confluence node
Yes
70
No
Node (I,J,K) is a confluence node
NFN(NF, K - 1) = I
Is K ≠ 2?
Are we looking at a confluence node on a level 3 stream?
Yes
70
No
NFS(NF) = J
70

Figure C.5 Program MAXEF

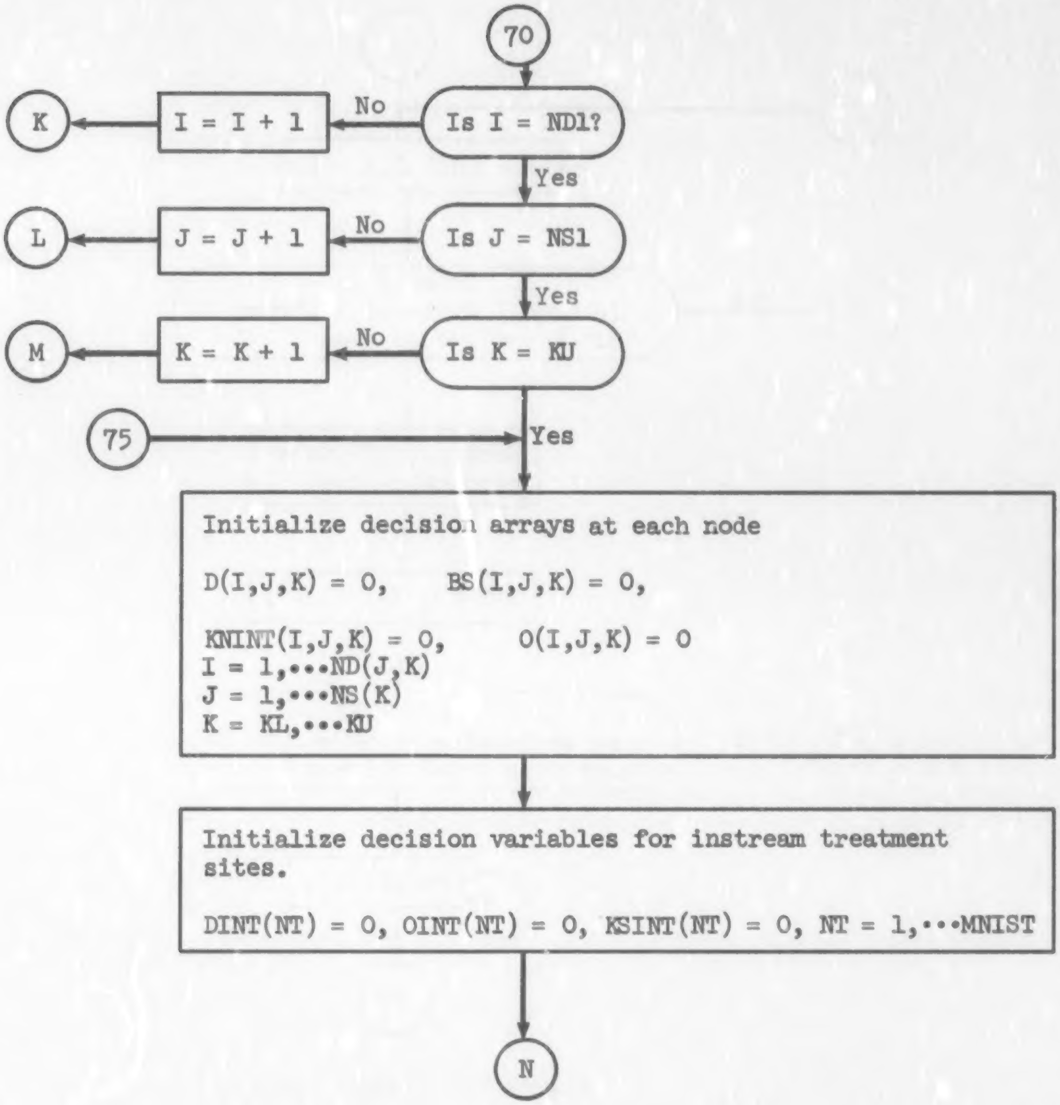
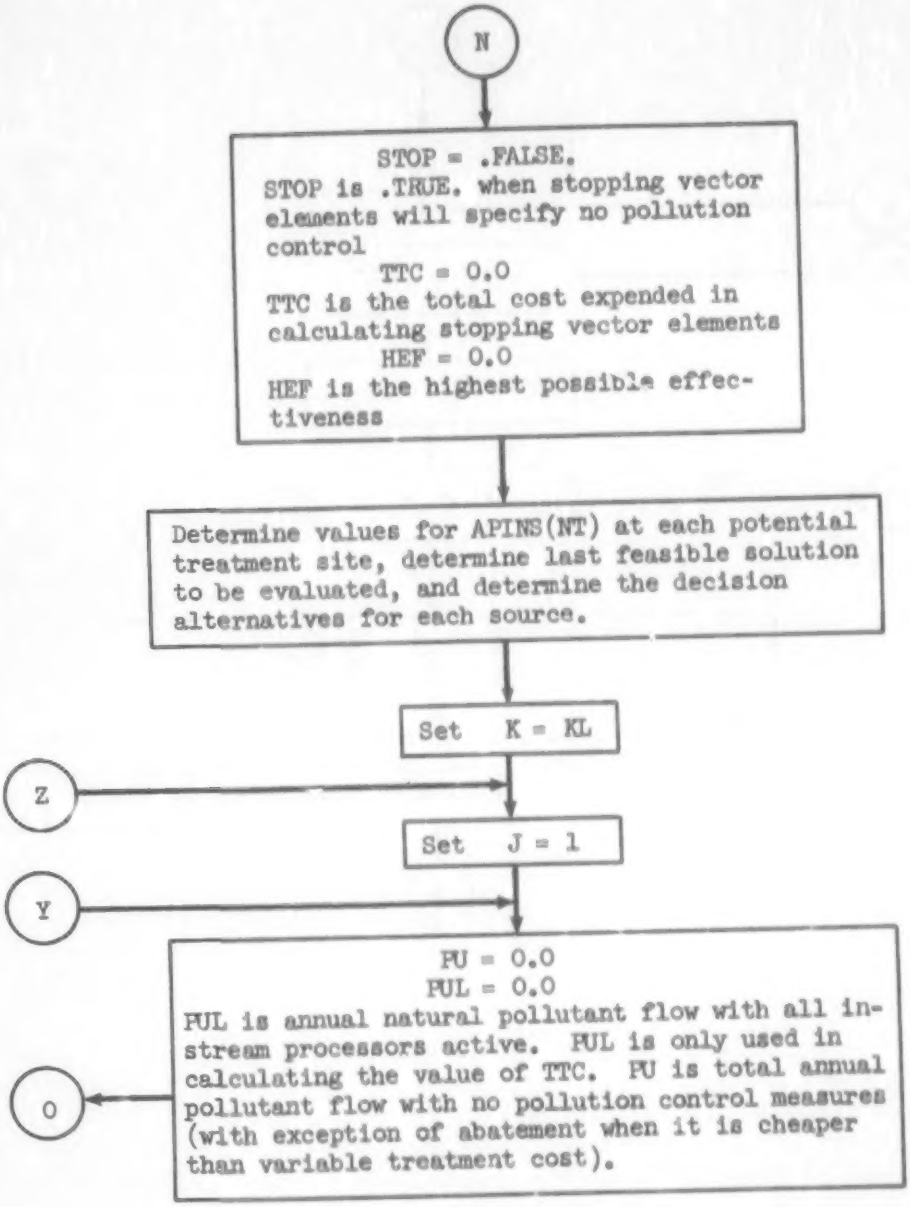

70
Is I = ND1?
No
I = I + 1
K
Yes
Is J = NS1
No
J = J + 1
L
Yes
Is K = KU
No
K = K + 1
M
75
Yes
Initialize decision arrays at each node
D(I,J,K) = 0, BS(I,J,K) = 0,
KNINT(I,J,K) = 0, O(I,J,K) = 0
I = 1,•••ND(J,K)
J = 1,•••NS(K)
K = KL,•••KU
Initialize decision variables for instream treatment sites.
DINT(NT) = 0, OINT(NT) = 0, KSINT(NT) = 0, NT = 1,•••MNIST
N

Figure C.5 Program MAXEF



STOP = .FALSE.
STOP is .TRUE. when stopping vector elements will specify no pollution control
TTC = 0.0
TTC is the total cost expended in calculating stopping vector elements
HEF = 0.0
HEF is the highest possible effectiveness

Determine values for APINS(NT) at each potential treatment site, determine last feasible solution to be evaluated, and determine the decision alternatives for each source.

Set K = KL

Set J = 1

PU = 0.0
PUL = 0.0
PUL is annual natural pollutant flow with all instream processors active. PUL is only used in calculating the value of TTC. PU is total annual pollutant flow with no pollution control measures (with exception of abatement when it is cheaper than variable treatment cost).

Figure C.5 Program MAXEF

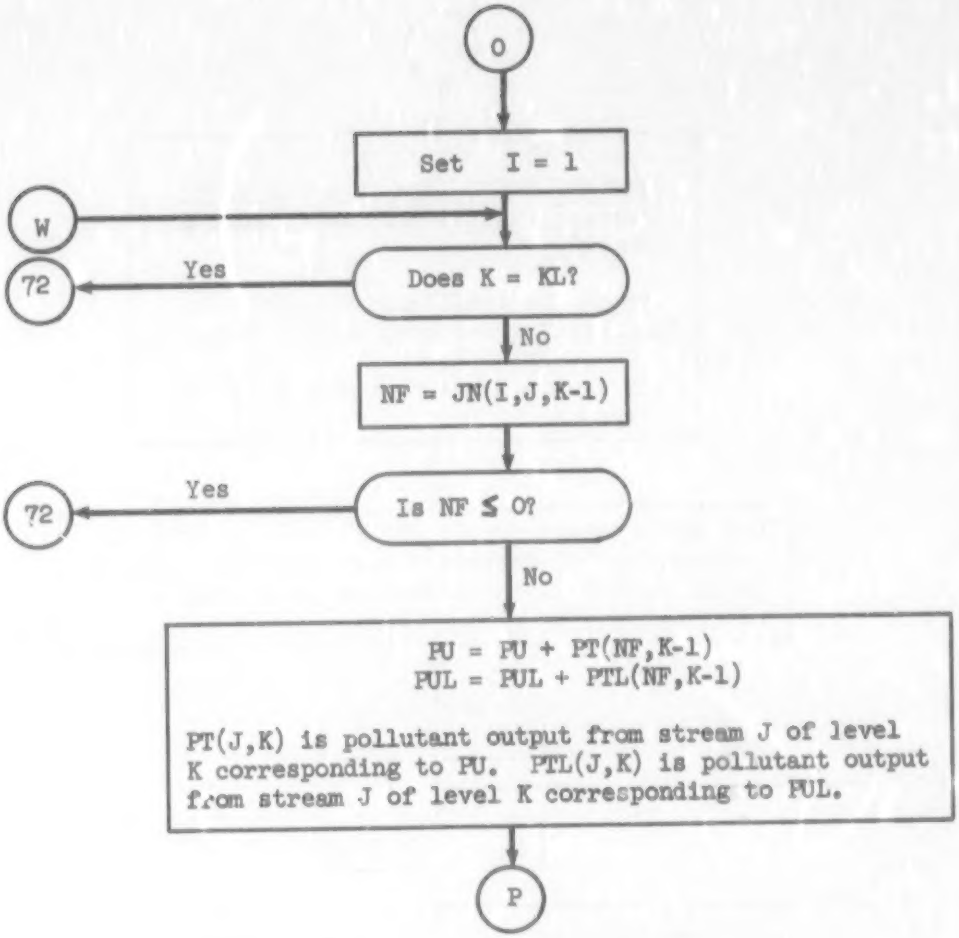

O
Set I = 1
W
Yes
72
Does K = KL?
No
NF = JN(I,J,K-1)
Yes
72
Is NF ≤ 0?
No
PU = PU + PT(NF,K-1)
PUL = PUL + PTL(NF,K-1)
PT(J,K) is pollutant output from stream J of level K corresponding to PU. PTL(J,K) is pollutant output from stream J of level K corresponding to PUL.
P

Figure C.5 Program MAXEF

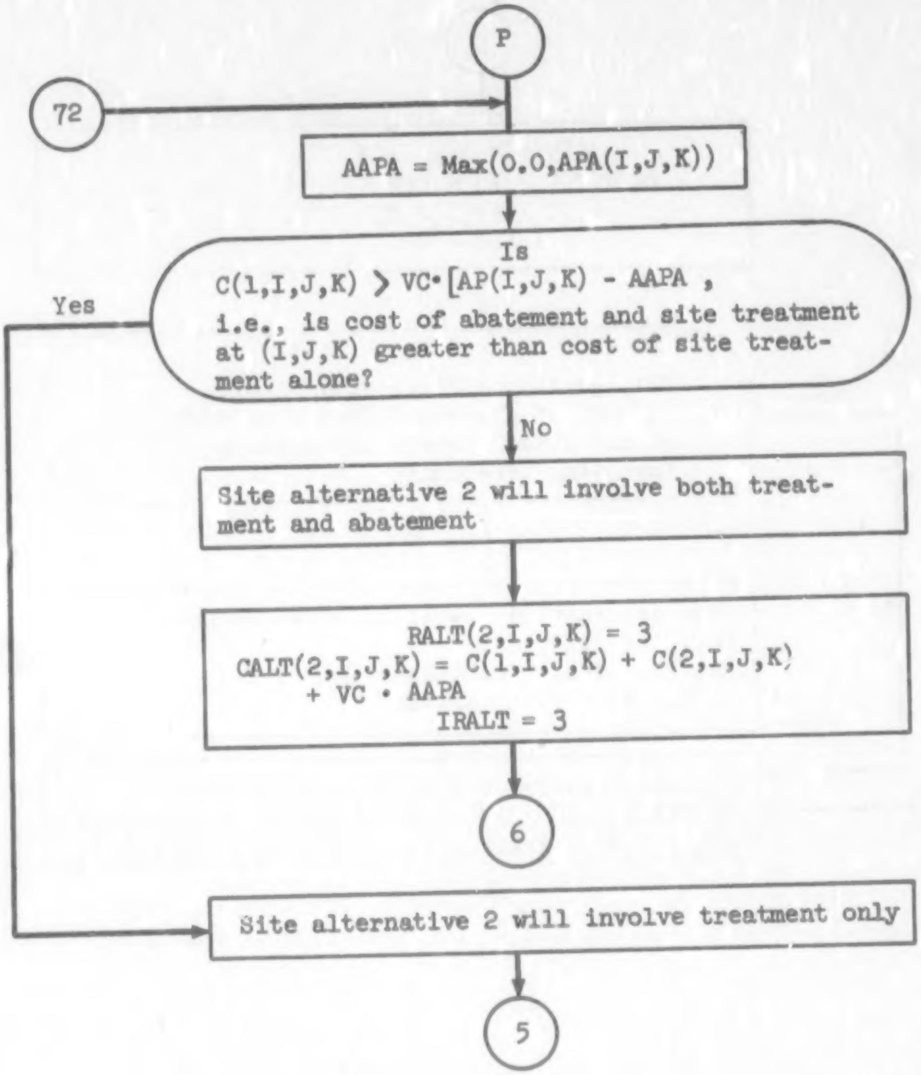

P
72
AAPA = Max(0.0,APA(I,J,K))
Is
C(1,I,J,K) > VC•[AP(I,J,K) - AAPA ,
i.e., is cost of abatement and site treatment at (I,J,K) greater than cost of site treatment alone?
Yes
No
Site alternative 2 will involve both treatment and abatement
RALT(2,I,J,K) = 3
CALT(2,I,J,K) = C(1,I,J,K) + C(2,I,J,K)
+ VC • AAPA
IRALT = 3
6
Site alternative 2 will involve treatment only
5

Figure C.5 Program MAXEF

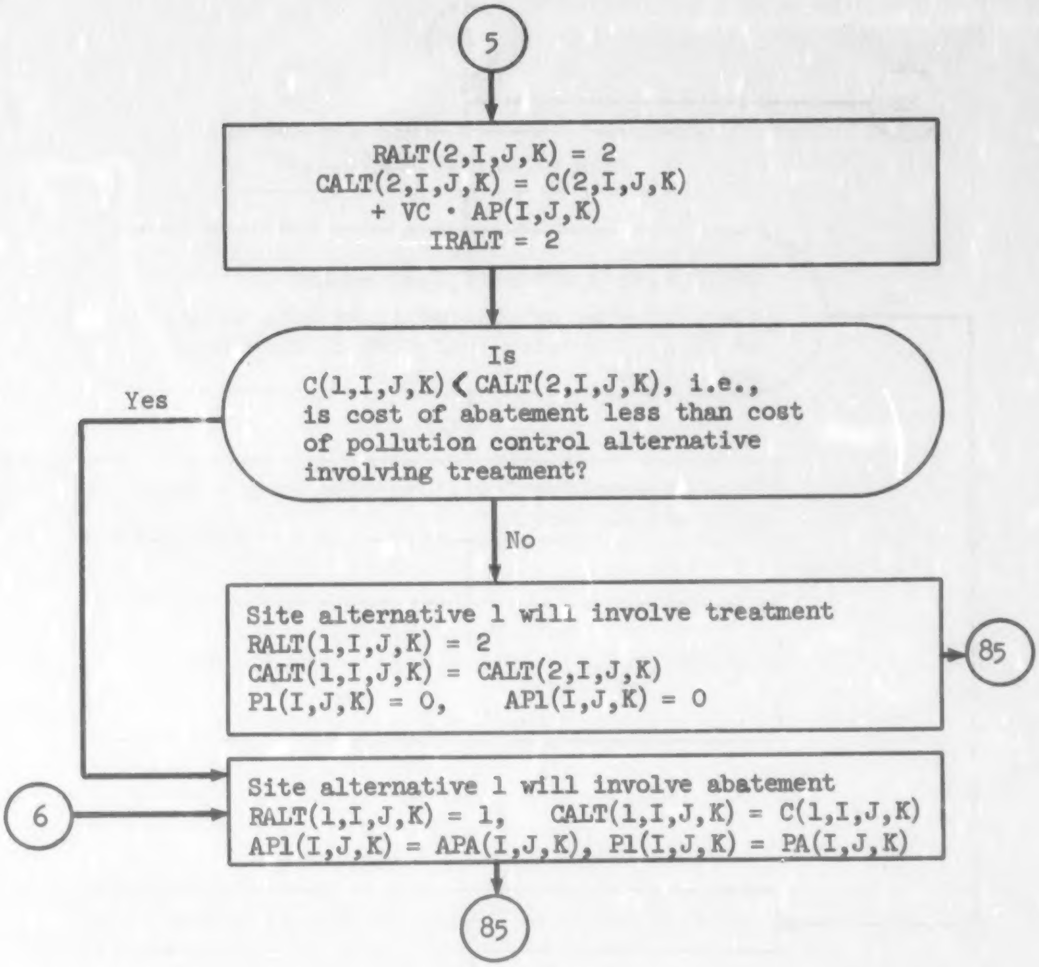

5
RALT(2,I,J,K) = 2
CALT(2,I,J,K) = C(2,I,J,K)
+ VC · AP(I,J,K)
IRALT = 2
Is
C(1,I,J,K) < CALT(2,I,J,K), i.e.,
is cost of abatement less than cost
of pollution control alternative
involving treatment?
Yes
No
Site alternative 1 will involve treatment
RALT(1,I,J,K) = 2
CALT(1,I,J,K) = CALT(2,I,J,K)
P1(I,J,K) = 0, AP1(I,J,K) = 0
85
6
Site alternative 1 will involve abatement
RALT(1,I,J,K) = 1, CALT(1,I,J,K) = C(1,I,J,K)
AP1(I,J,K) = APA(I,J,K), P1(I,J,K) = PA(I,J,K)
85

Figure C.5 Program MAXEF

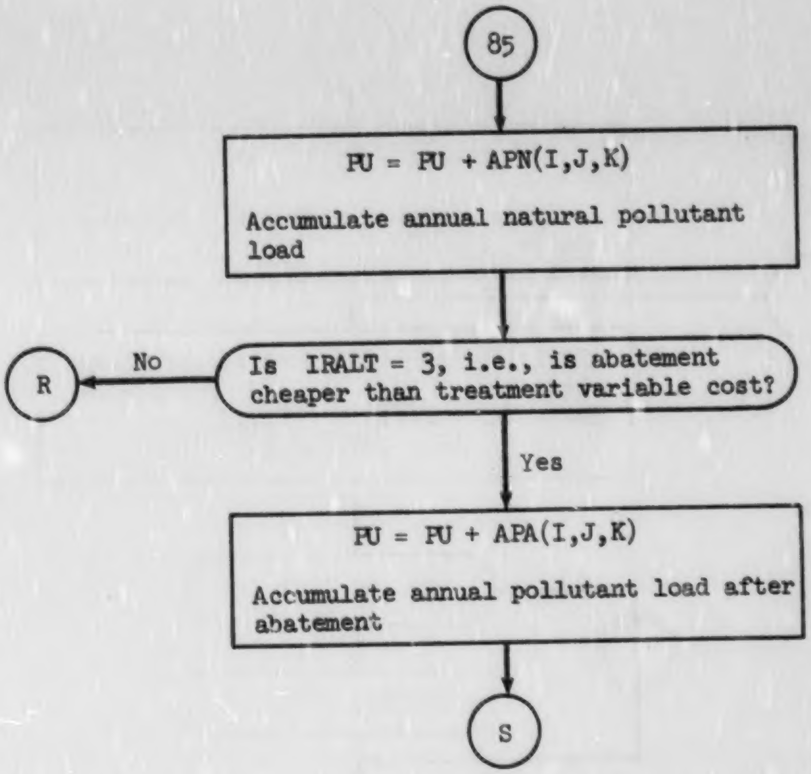

85
PU = PU + APN(I,J,K)
Accumulate annual natural pollutant load
Is IRALT = 3, i.e., is abatement cheaper than treatment variable cost?
No
R
Yes
PU = PU + APA(I,J,K)
Accumulate annual pollutant load after abatement
S

Figure C.5 Program MAXEF

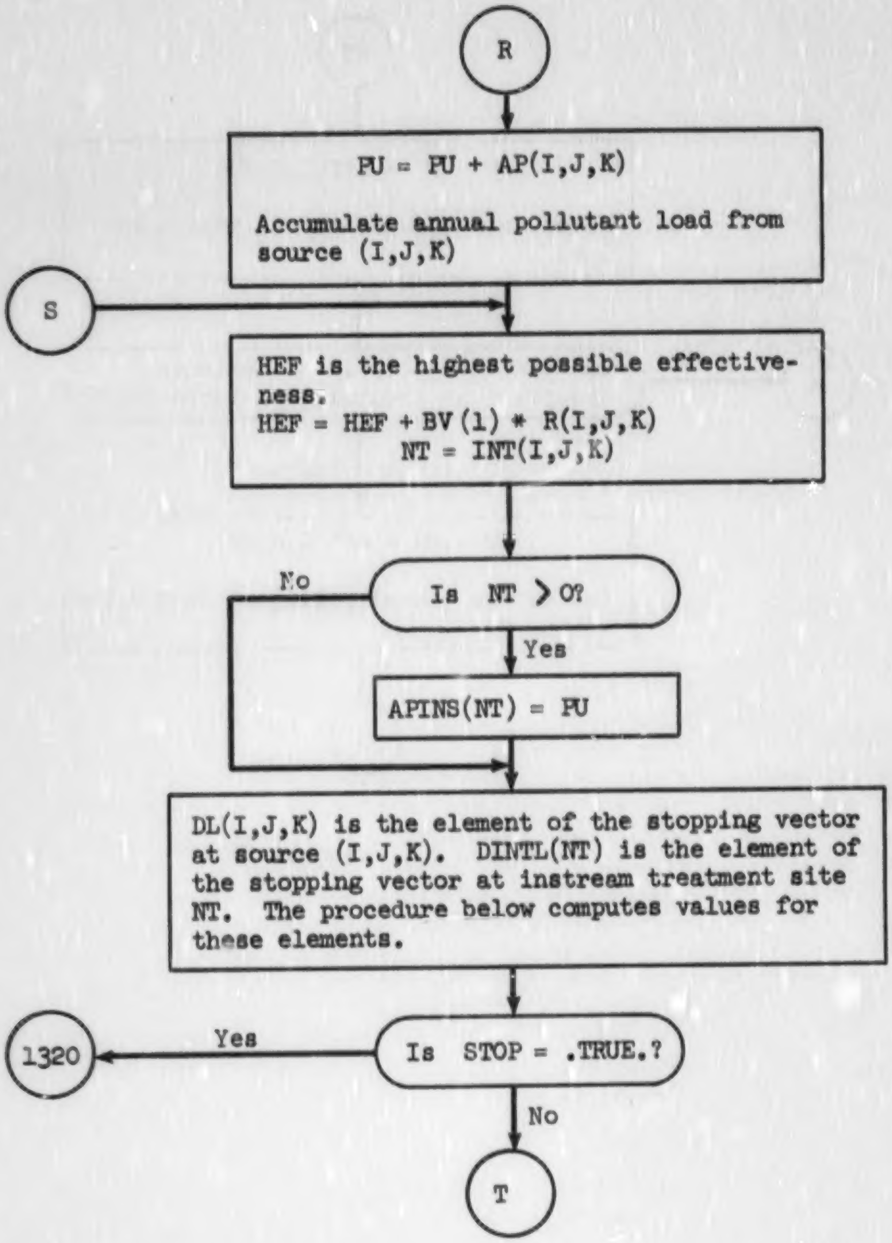

R
PU = PU + AP(I,J,K)
Accumulate annual pollutant load from source (I,J,K)
S
HEF is the highest possible effectiveness.
HEF = HEF + BV (1) * R(I,J,K)
NT = INT(I,J,K)
Is NT > 0?
No
Yes
APINS(NT) = PU
DL(I,J,K) is the element of the stopping vector at source (I,J,K). DINTL(NT) is the element of the stopping vector at instream treatment site NT. The procedure below computes values for these elements.
Is STOP = .TRUE.?
Yes
1320
No
T

Figure C.5 Program MAXEF

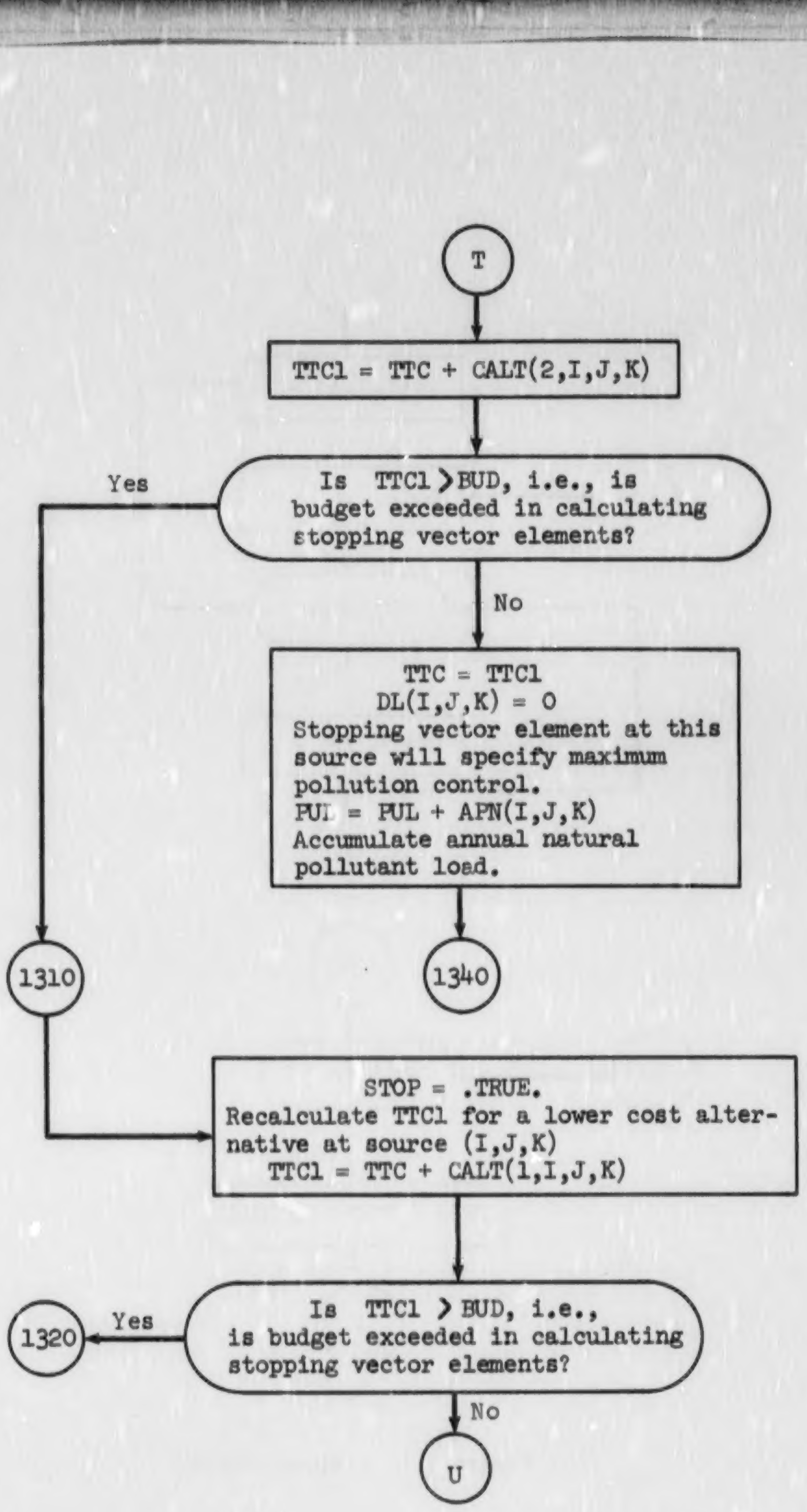

T
TTC1 = TTC + CALT(2,I,J,K)
Is TTC1 > BUD, i.e., is budget exceeded in calculating stopping vector elements?
Yes
No
TTC = TTC1
DL(I,J,K) = 0
Stopping vector element at this source will specify maximum pollution control.
PUL = PUL + APN(I,J,K)
Accumulate annual natural pollutant load.
1310
1340
STOP = .TRUE.
Recalculate TTC1 for a lower cost alternative at source (I,J,K)
TTC1 = TTC + CALT(1,I,J,K)
Is TTC1 > BUD, i.e., is budget exceeded in calculating stopping vector elements?
1320
Yes
No
U

Figure C.5 Program MAXEF

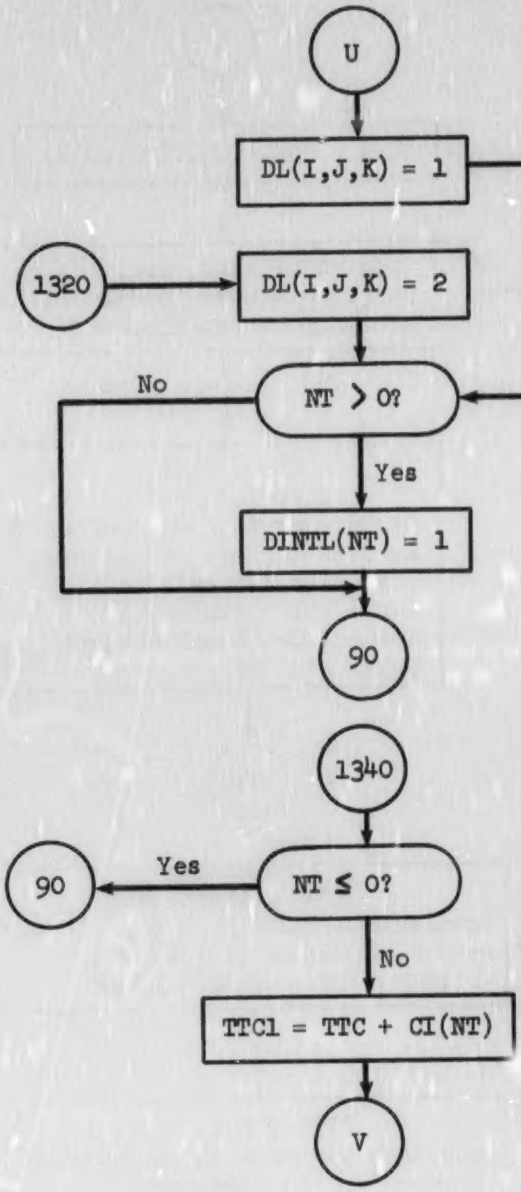

U
DL(I,J,K) = 1
1320
DL(I,J,K) = 2
No
NT > 0?
Yes
DINTL(NT) = 1
90
1340
90
Yes
NT ≤ 0?
No
TTC1 = TTC + CI(NT)
V

Figure C.5 Program MAXEF

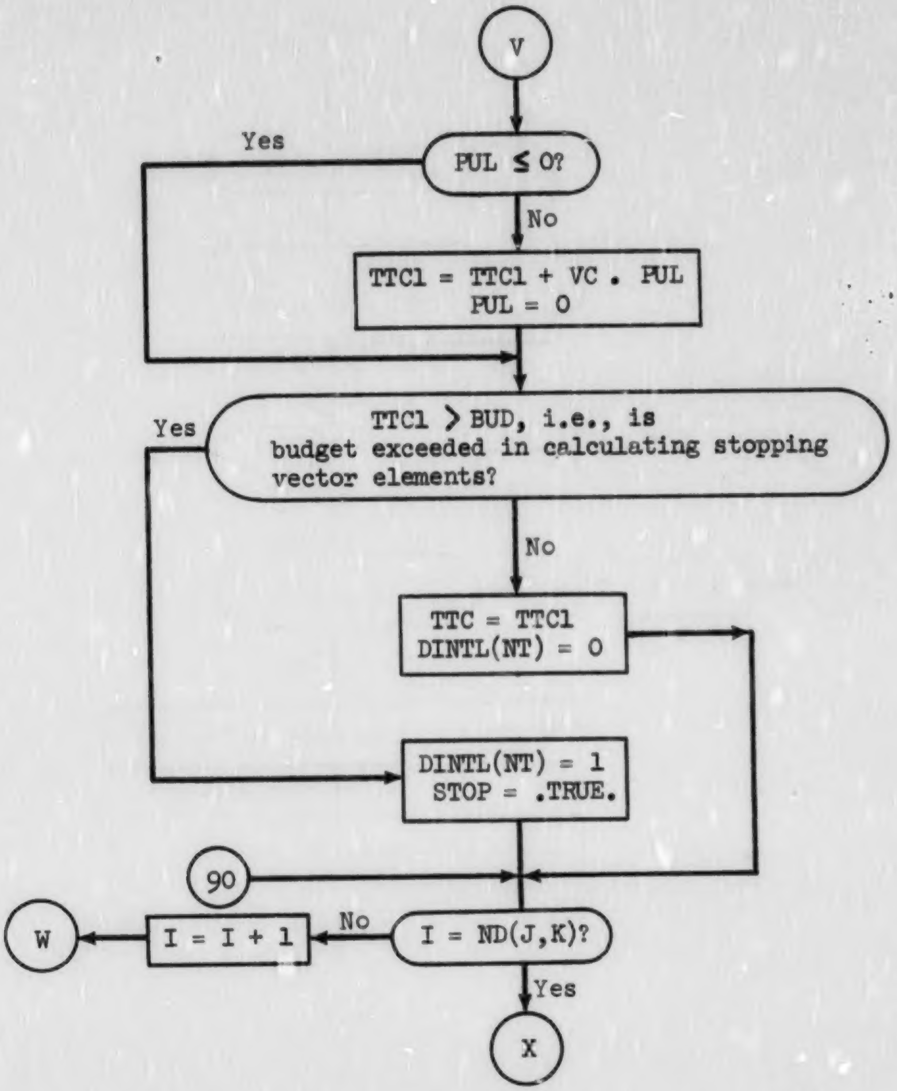

V
PUL ≤ 0?
Yes
No
TTC1 = TTC1 + VC . PUL
PUL = 0
TTC1 > BUD, i.e., is budget exceeded in calculating stopping vector elements?
Yes
No
TTC = TTC1
DINTL(NT) = 0
DINTL(NT) = 1
STOP = .TRUE.
90
W
I = I + 1
No
I = ND(J,K)?
Yes
X

Figure C.5 Program MAXEF

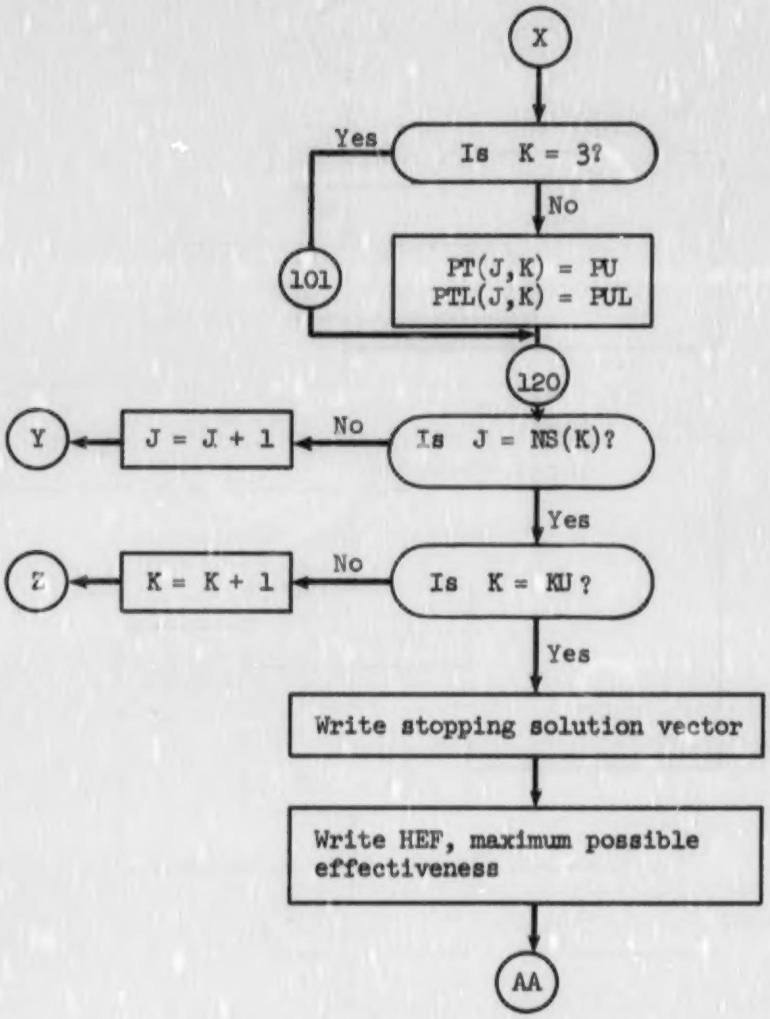

X
Is K = 3?
Yes
No
PT(J,K) = PU
PTL(J,K) = PUL
101
120
Is J = NS(K)?
No
J = J + 1
Y
Yes
Is K = KU ?
No
K = K + 1
Z
Yes
Write stopping solution vector
Write HEF, maximum possible effectiveness
AA

Figure C.5 Program MAXEF

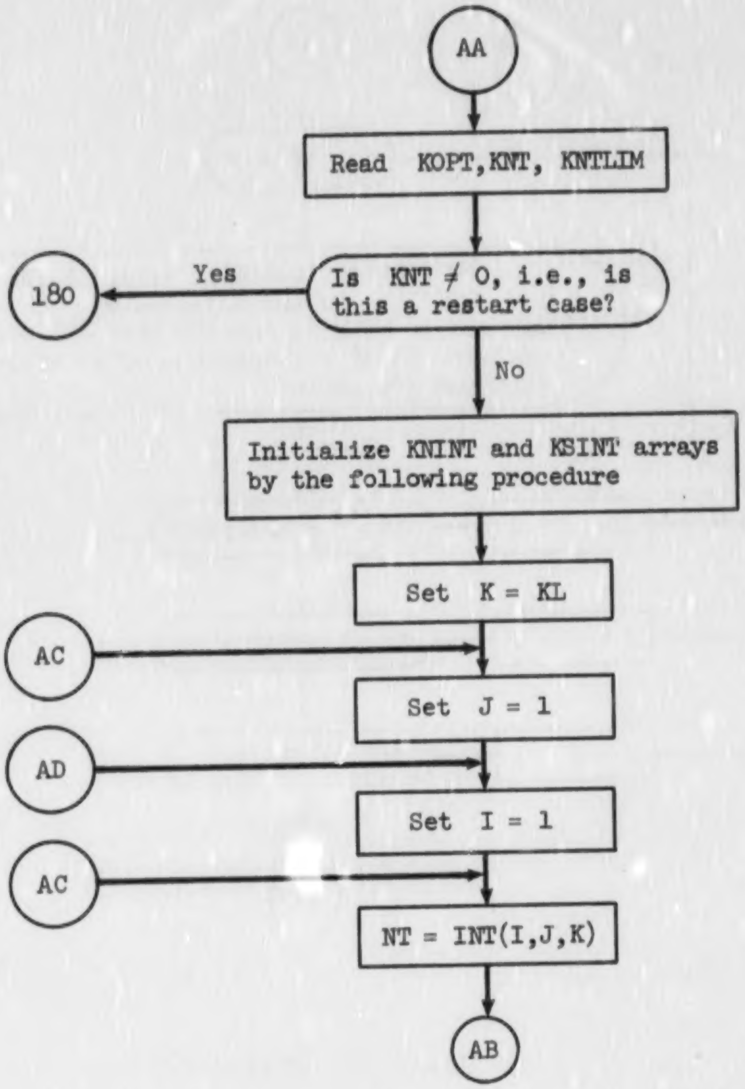

AA
Read KOPT,KNT, KNTLIM
Is KNT ≠ 0, i.e., is this a restart case?
Yes
180
No
Initialize KNINT and KSINT arrays by the following procedure
Set K = KL
AC
Set J = 1
AD
Set I = 1
AC
NT = INT(I,J,K)
AB

Figure C.5 Program MAXEF

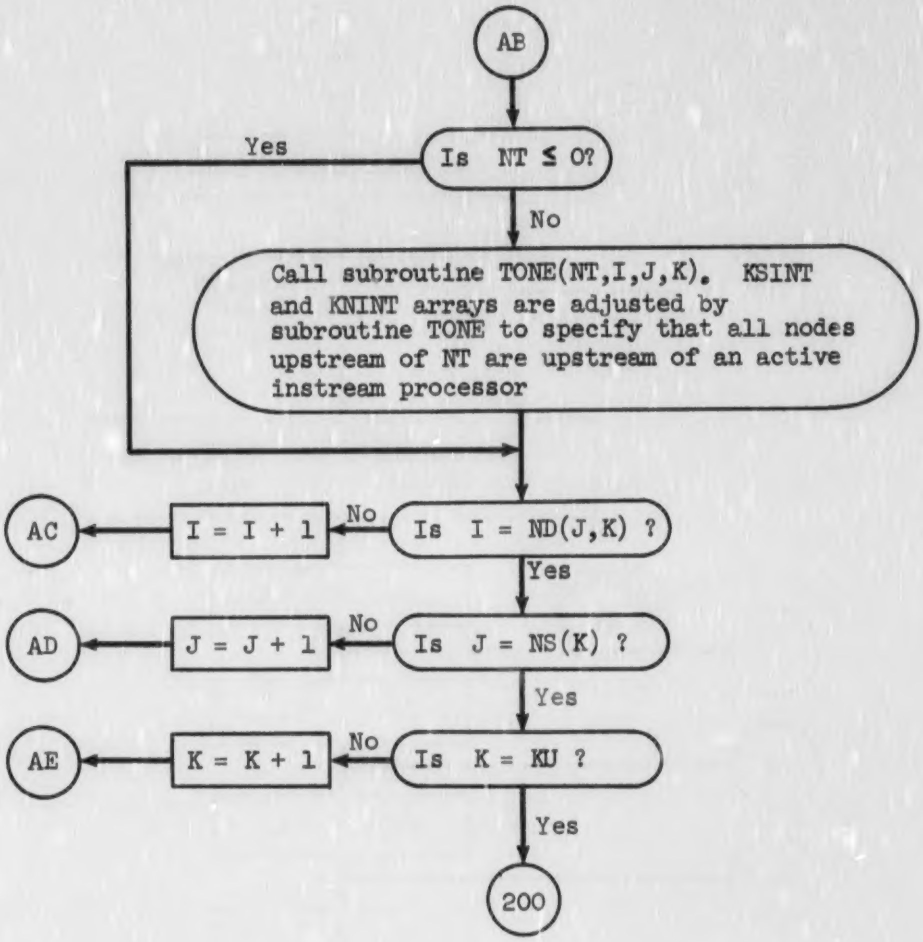

AB
Is NT ≤ 0?
Yes
No
Call subroutine TONE(NT,I,J,K). KSINT and KNINT arrays are adjusted by subroutine TONE to specify that all nodes upstream of NT are upstream of an active instream processor
Is I = ND(J,K) ?
No
I = I + 1
AC
Yes
Is J = NS(K) ?
No
J = J + 1
AD
Yes
Is K = KU ?
No
K = K + 1
AE
Yes
200

Figure C.5 Program MAXEF

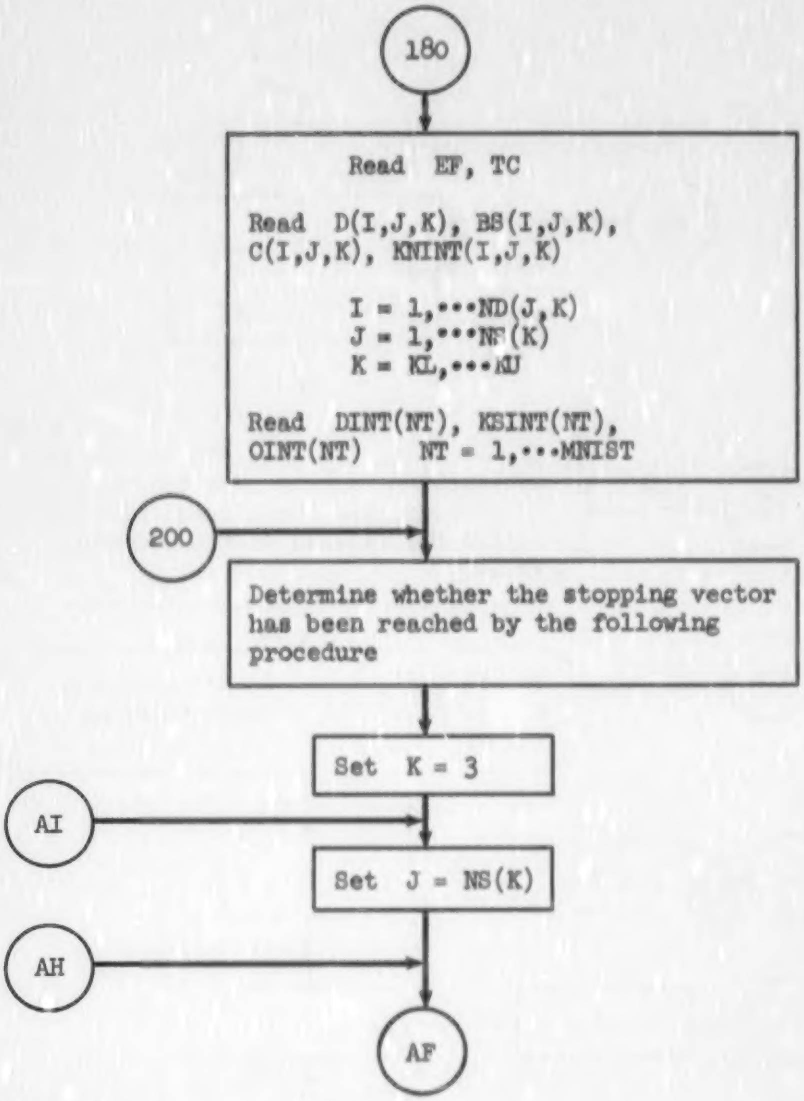

180
Read EF, TC
Read D(I,J,K), BS(I,J,K), C(I,J,K), KNINT(I,J,K)
I = 1,•••ND(J,K)
J = 1,•••NF(K)
K = KL,•••KU
Read DINT(NT), KSINT(NT), OINT(NT) NT = 1,•••MNIST
200
Determine whether the stopping vector has been reached by the following procedure
Set K = 3
AI
Set J = NS(K)
AH
AF

Figure C.5 Program MAXEF

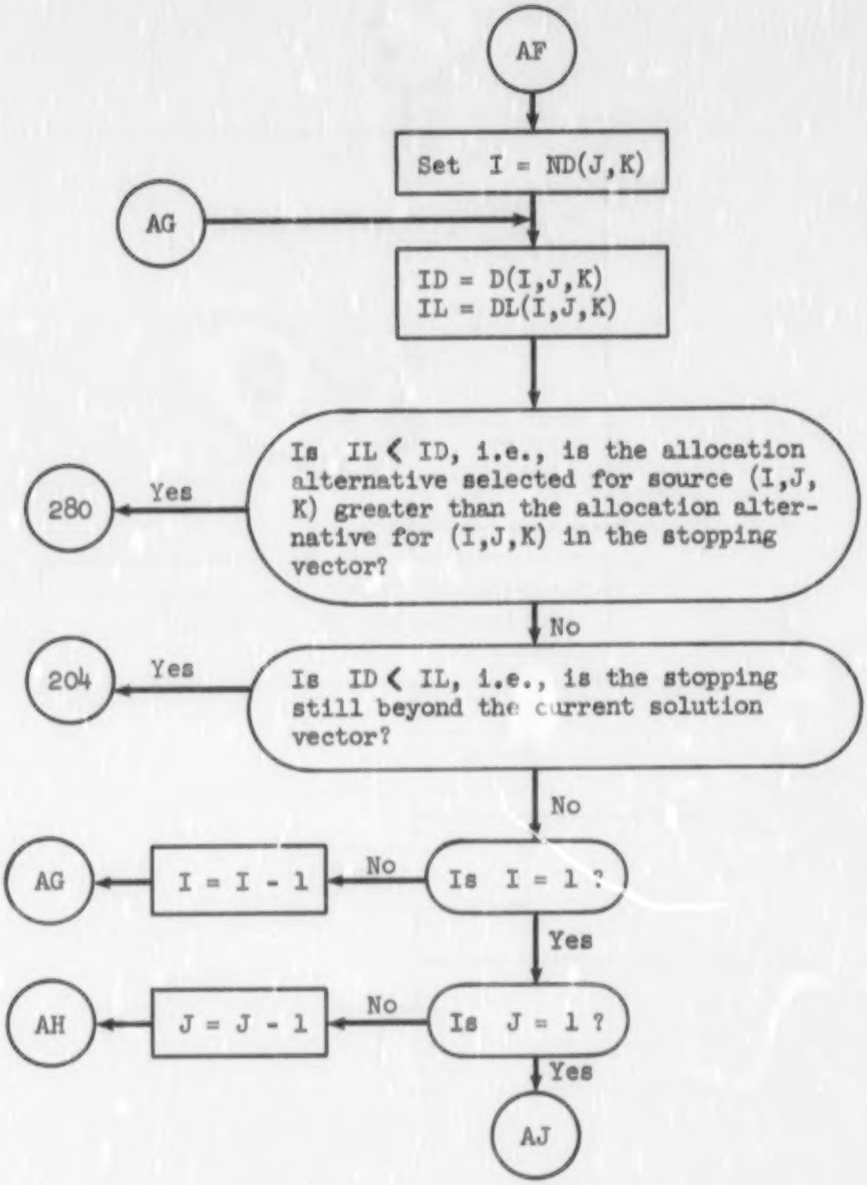

AF
Set I = ND(J,K)
AG
ID = D(I,J,K)
IL = DL(I,J,K)
Is IL < ID, i.e., is the allocation alternative selected for source (I,J, K) greater than the allocation alternative for (I,J,K) in the stopping vector?
Yes
280
No
Is ID < IL, i.e., is the stopping still beyond the current solution vector?
Yes
204
No
Is I = 1 ?
No
I = I - 1
AG
Yes
Is J = 1 ?
No
J = J - 1
AH
Yes
AJ

Figure C.5 Program MAXEF

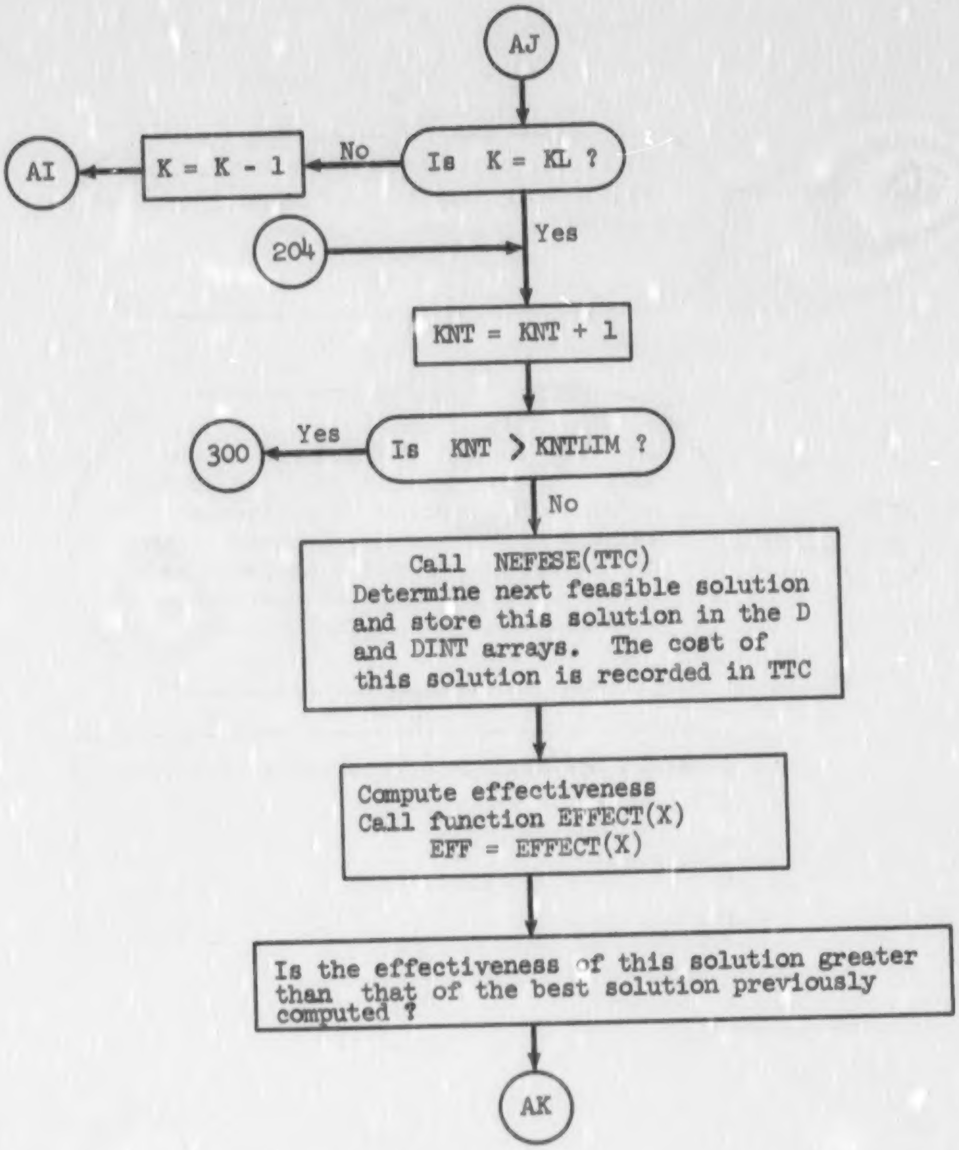

AJ
Is K = KL ?
No
K = K - 1
AI
Yes
204
KNT = KNT + 1
Is KNT > KNTLIM ?
Yes
300
No
Call NEFESE(TTC)
Determine next feasible solution and store this solution in the D and DINT arrays. The cost of this solution is recorded in TTC
Compute effectiveness
Call function EFFECT(X)
EFF = EFFECT(X)
Is the effectiveness of this solution greater than that of the best solution previously computed ?
AK

Figure C.5 Program MAXEF

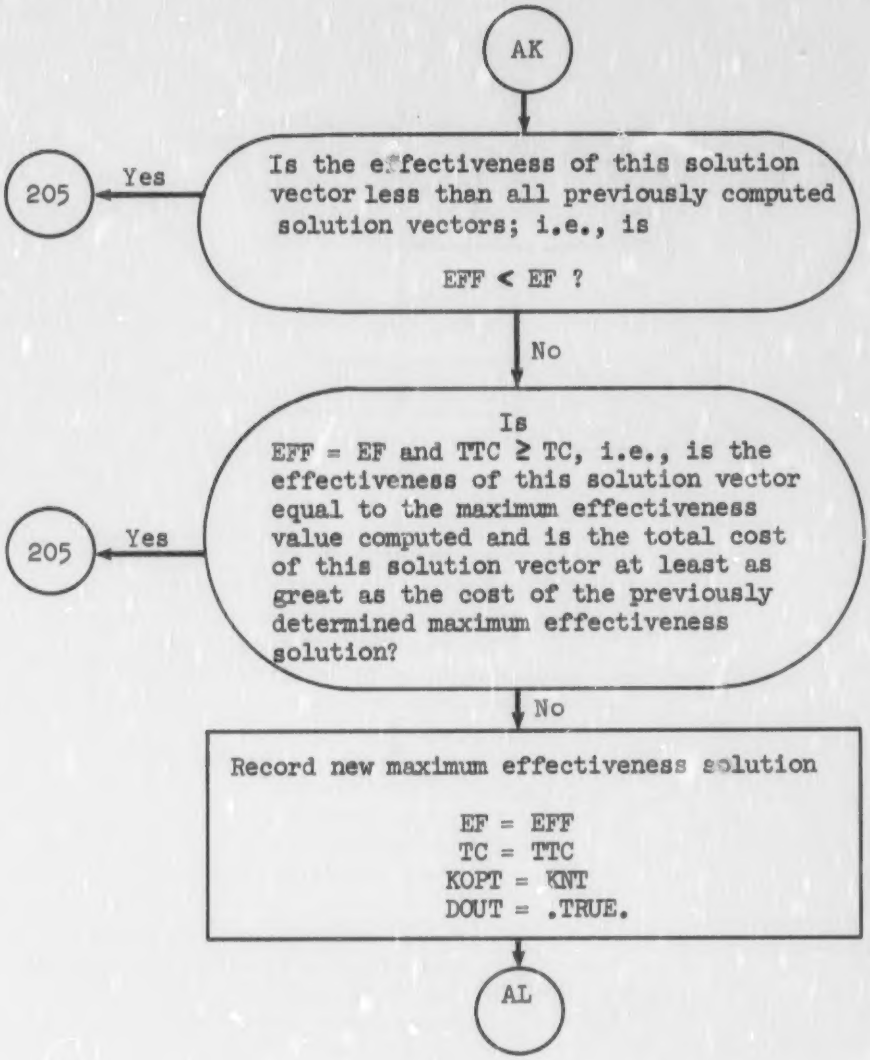

AK
Is the effectiveness of this solution vector less than all previously computed solution vectors; i.e., is EFF < EF ?
Yes
205
No
Is EFF = EF and TTC ≥ TC, i.e., is the effectiveness of this solution vector equal to the maximum effectiveness value computed and is the total cost of this solution vector at least as great as the cost of the previously determined maximum effectiveness solution?
Yes
205
No
Record new maximum effectiveness solution
EF = EFF
TC = TTC
KOPT = KNT
DOUT = .TRUE.
AL

Figure C.5 Program MAXEF

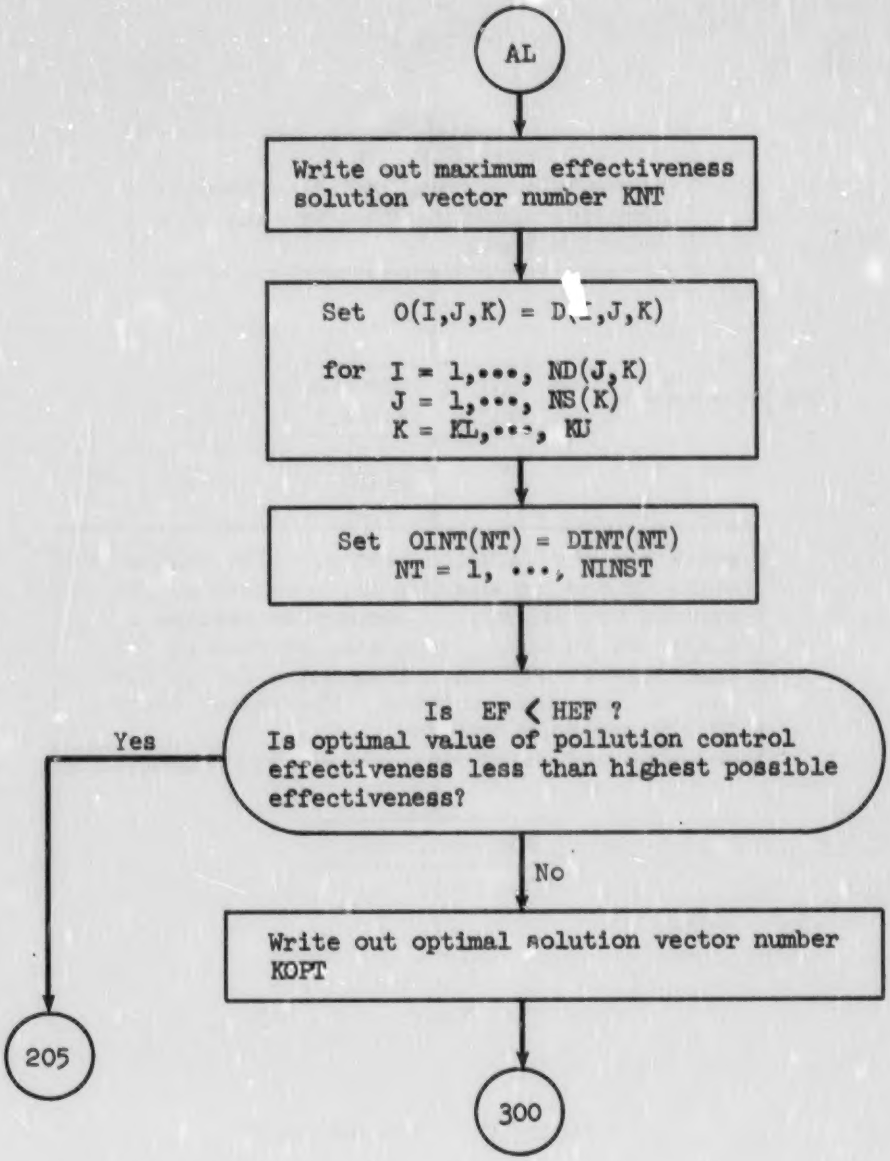

AL
Write out maximum effectiveness solution vector number KNT
Set O(I,J,K) = D(I,J,K)
for I = 1,•••, ND(J,K)
J = 1,•••, NS(K)
K = KL,•••, KU
Set OINT(NT) = DINT(NT)
NT = 1, •••, NINST
Is EF < HEF ?
Is optimal value of pollution control effectiveness less than highest possible effectiveness?
Yes
No
Write out optimal solution vector number KOPT
205
300

Figure C.5 Program MAXEF

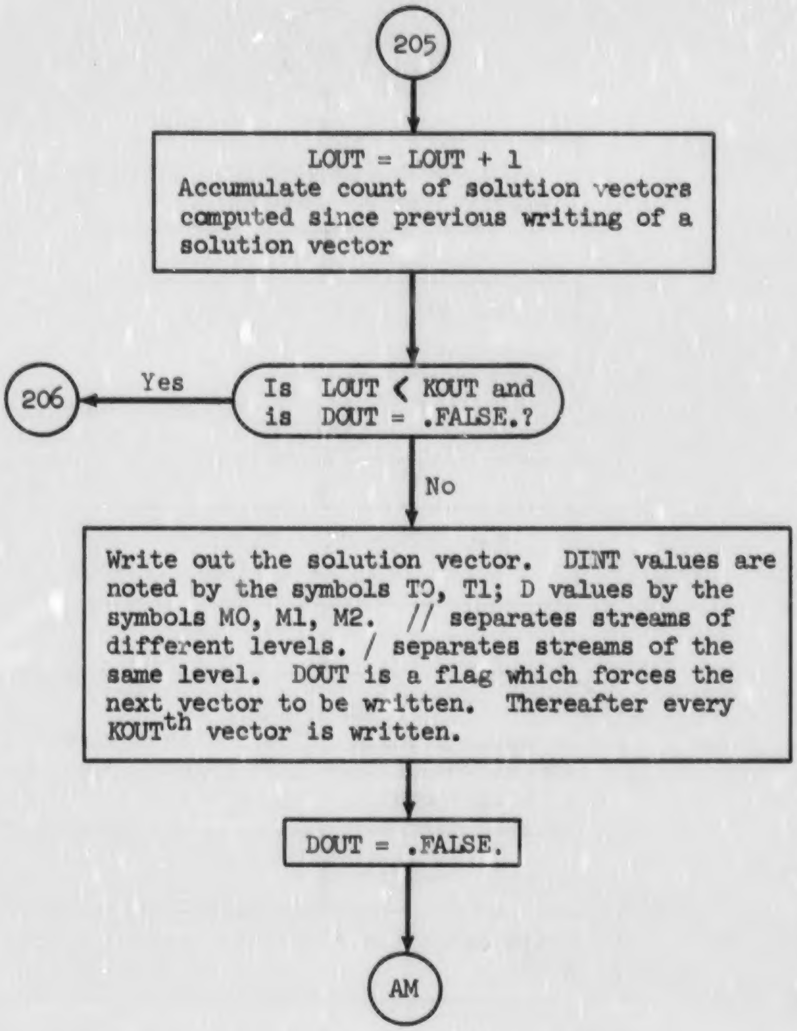

205
LOUT = LOUT + 1
Accumulate count of solution vectors computed since previous writing of a solution vector
206
Yes
Is LOUT < KOUT and is DOUT = .FALSE.?
No
Write out the solution vector. DINT values are noted by the symbols T0, T1; D values by the symbols M0, M1, M2. // separates streams of different levels. / separates streams of the same level. DOUT is a flag which forces the next vector to be written. Thereafter every KOUTth vector is written.
DOUT = .FALSE.
AM

Figure C.5 Program MAXEF

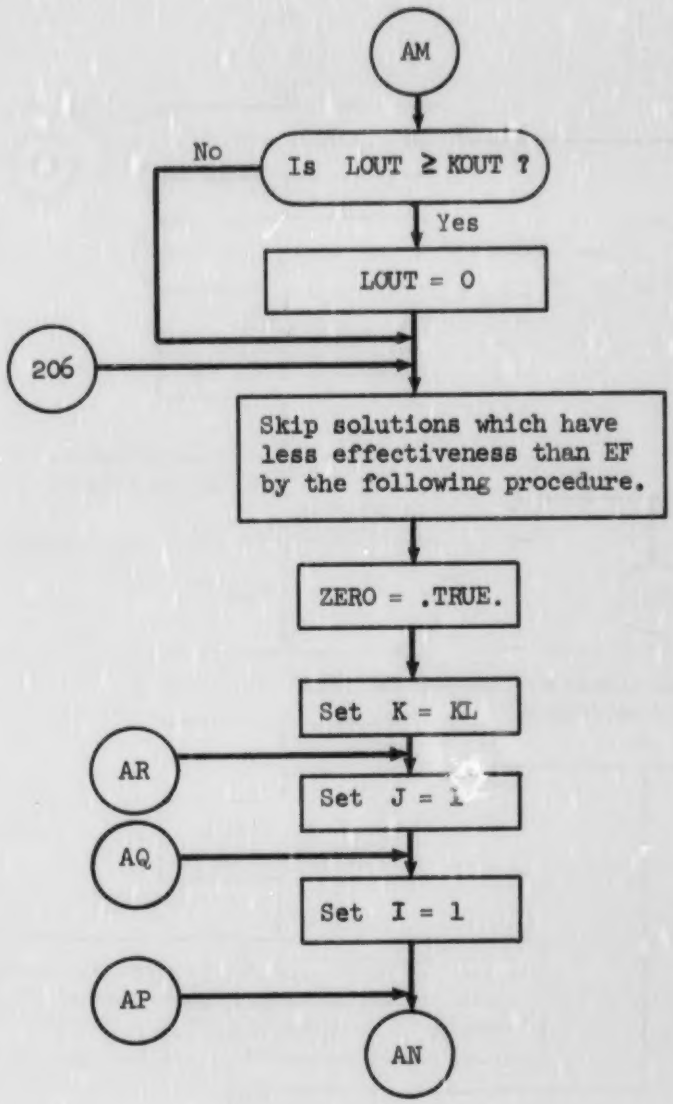

AM
No
Is LOUT ≥ KOUT ?
Yes
LOUT = 0
206
Skip solutions which have less effectiveness than EF by the following procedure.
ZERO = .TRUE.
Set K = KL
AR
Set J = 1
AQ
Set I = 1
AP
AN

Figure C.5 Program MAXEF

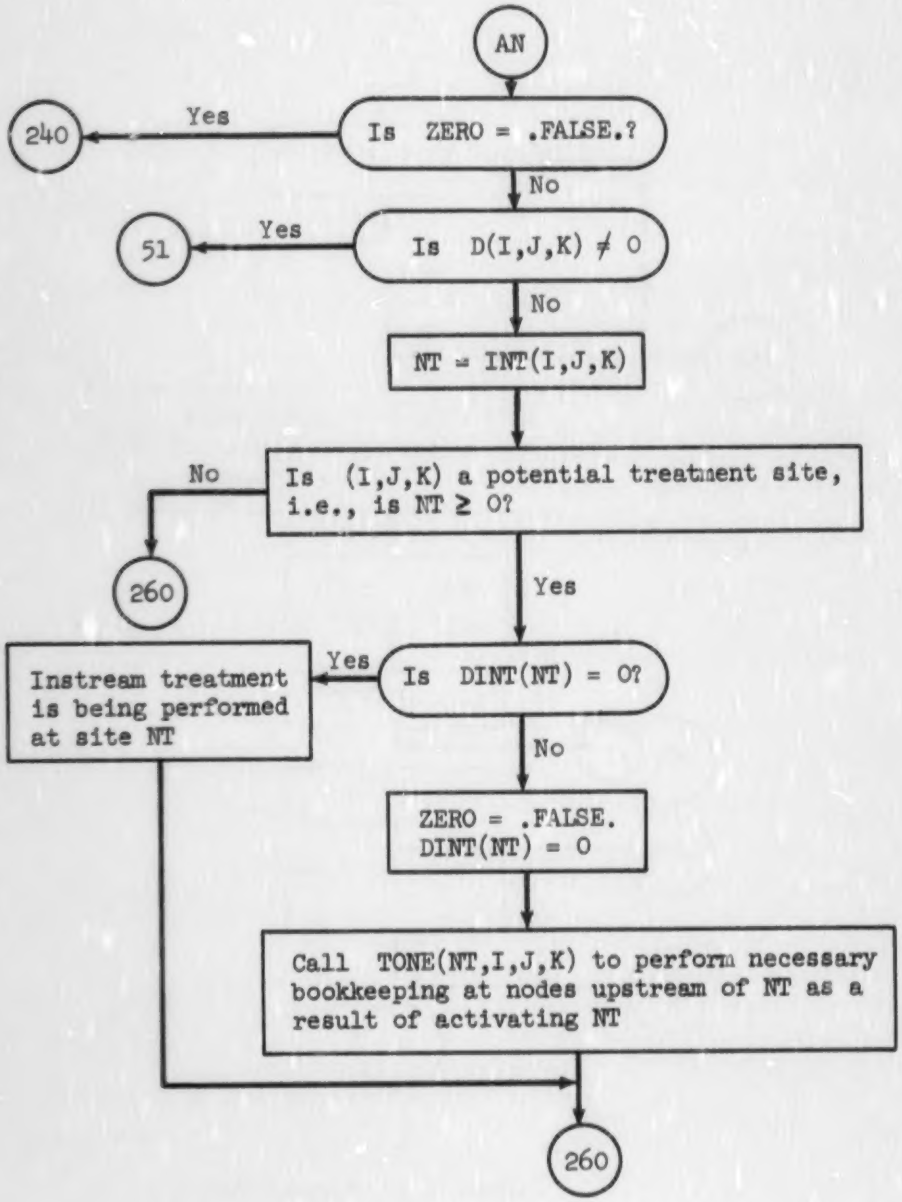

AN
Is ZERO = .FALSE.?
Yes
240
No
Is D(I,J,K) ≠ 0
Yes
51
No
NT = INT(I,J,K)
Is (I,J,K) a potential treatment site, i.e., is NT ≥ 0?
No
260
Yes
Is DINT(NT) = 0?
Yes
Instream treatment is being performed at site NT
No
ZERO = .FALSE.
DINT(NT) = 0
Call TONE(NT,I,J,K) to perform necessary bookkeeping at nodes upstream of NT as a result of activating NT
260

Figure C.5 Program MAXEF

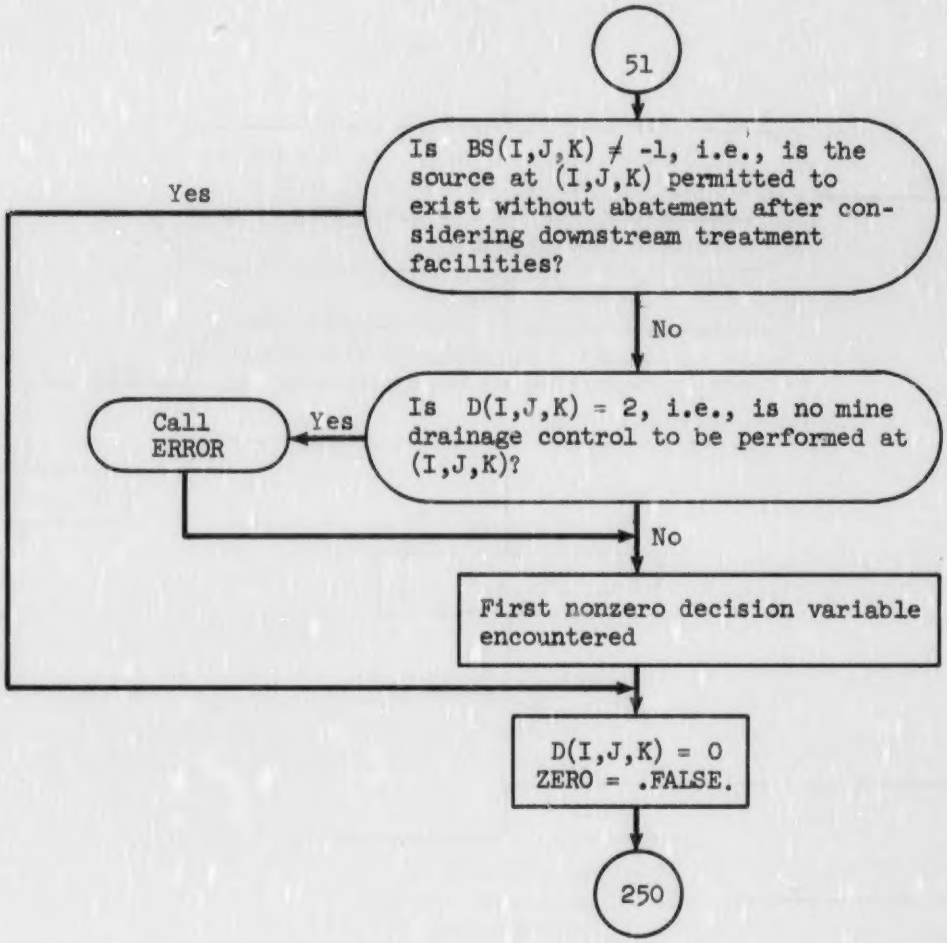

51
Is BS(I,J,K) ≠ -1, i.e., is the source at (I,J,K) permitted to exist without abatement after considering downstream treatment facilities?
Yes
No
Is D(I,J,K) = 2, i.e., is no mine drainage control to be performed at (I,J,K)?
Yes
Call ERROR
No
First nonzero decision variable encountered
D(I,J,K) = 0
ZERO = .FALSE.
250

Figure C.5 Program MAXEF

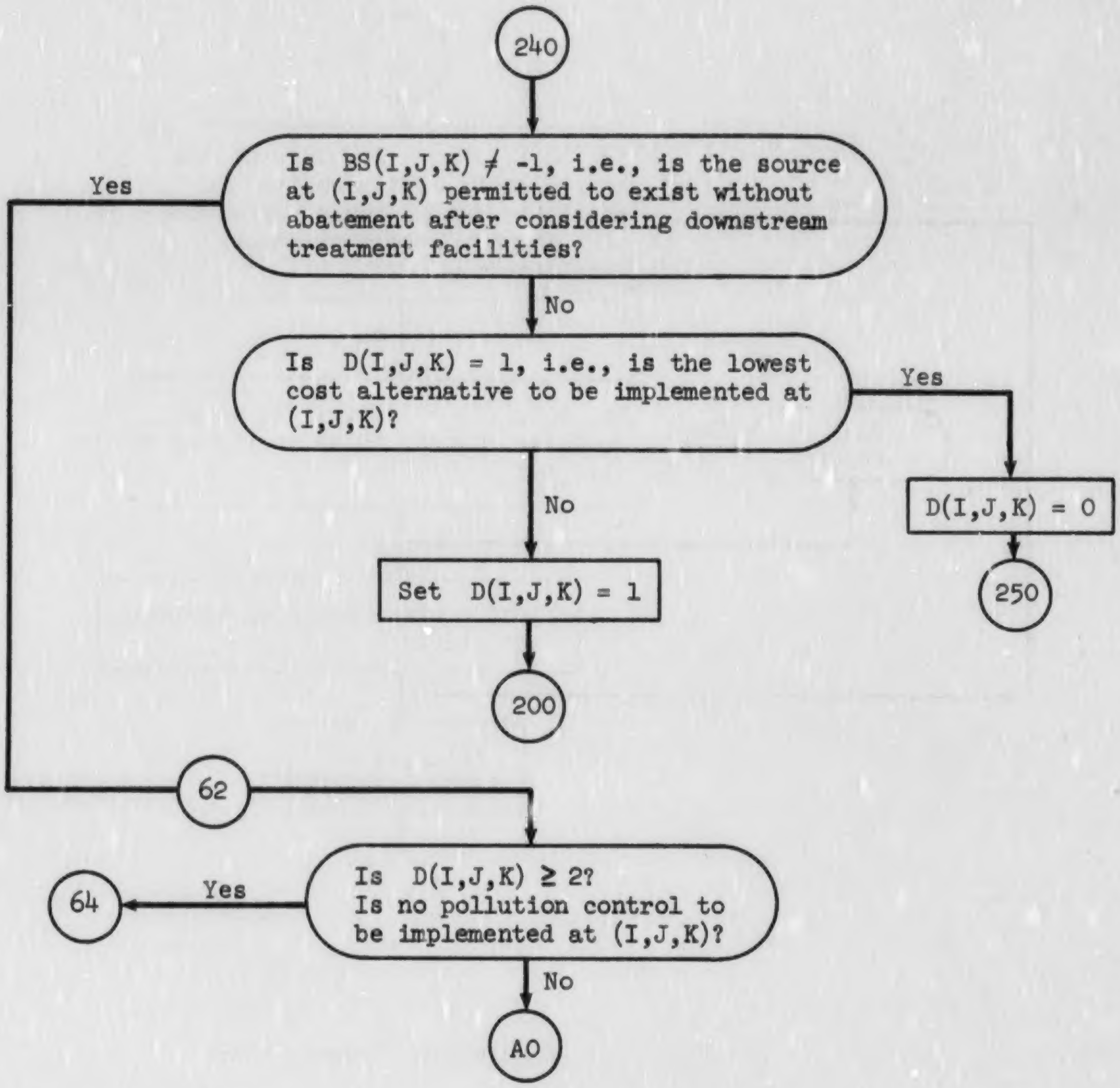

240
Is BS(I,J,K) ≠ -1, i.e., is the source at (I,J,K) permitted to exist without abatement after considering downstream treatment facilities?
Yes
No
Is D(I,J,K) = 1, i.e., is the lowest cost alternative to be implemented at (I,J,K)?
Yes
No
D(I,J,K) = 0
250
Set D(I,J,K) = 1
200
62
Is D(I,J,K) ≥ 2?
Is no pollution control to be implemented at (I,J,K)?
Yes
64
No
AO

Figure C.5 Program MAXEF

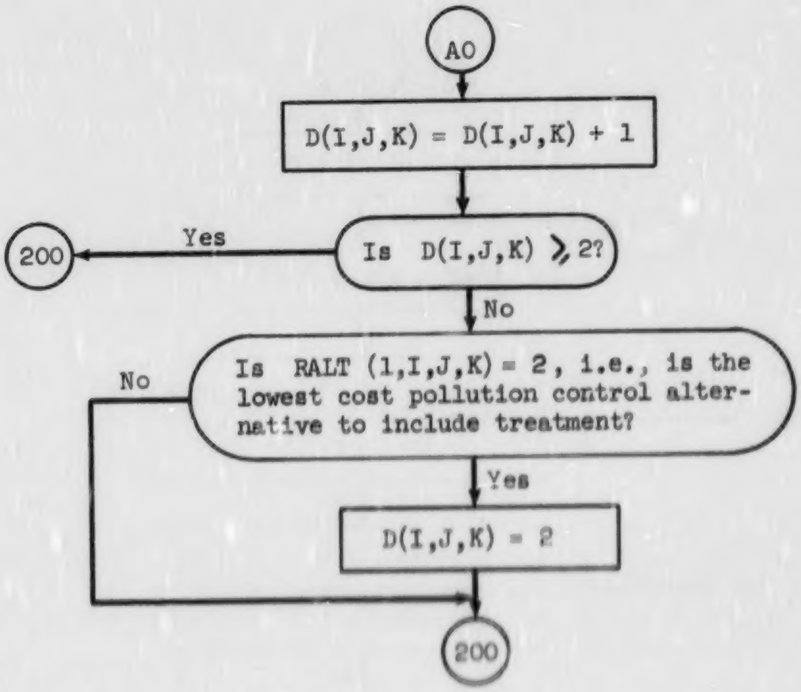

AO
D(I,J,K) = D(I,J,K) + 1
Is D(I,J,K) ≥ 2?
Yes
200
No
Is RALT (1,I,J,K) = 2, i.e., is the lowest cost pollution control alternative to include treatment?
No
Yes
D(I,J,K) = 2
200

Figure C.5 Program MAXEF

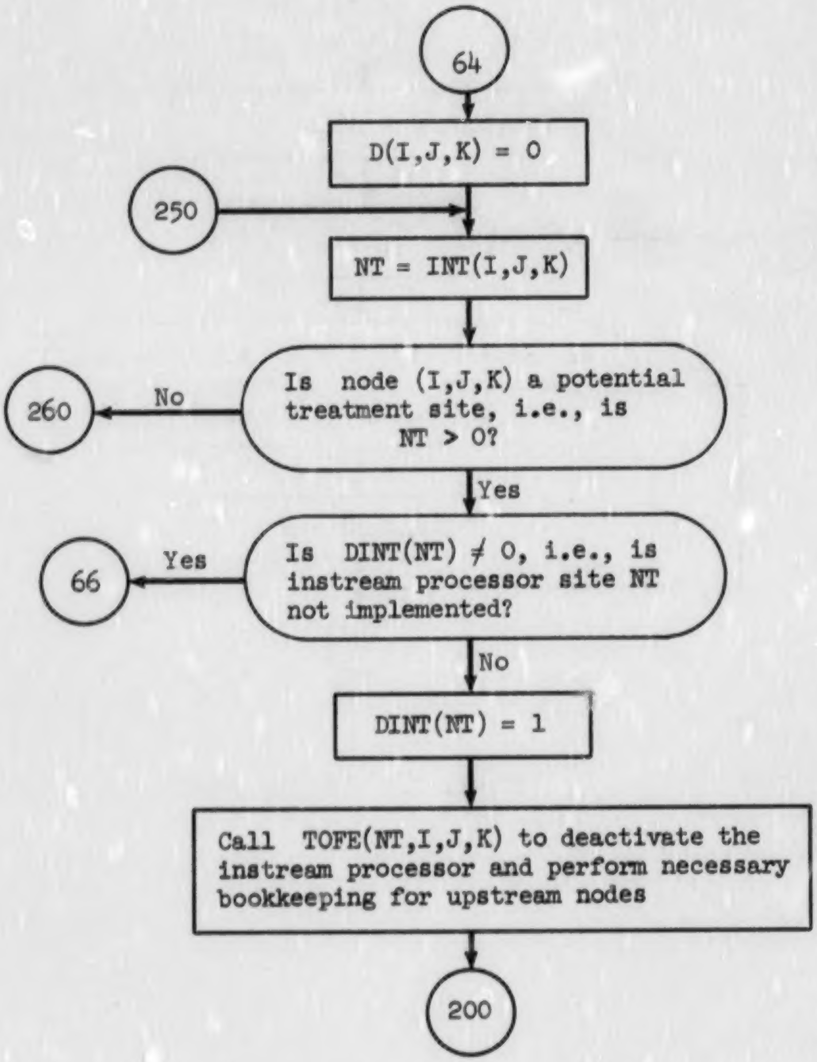

64
D(I,J,K) = 0
250
NT = INT(I,J,K)
Is node (I,J,K) a potential treatment site, i.e., is NT > 0?
No
260
Yes
Is DINT(NT) ≠ 0, i.e., is instream processor site NT not implemented?
Yes
66
No
DINT(NT) = 1
Call TOFE(NT,I,J,K) to deactivate the instream processor and perform necessary bookkeeping for upstream nodes
200

Figure C.5 Program MAXEF

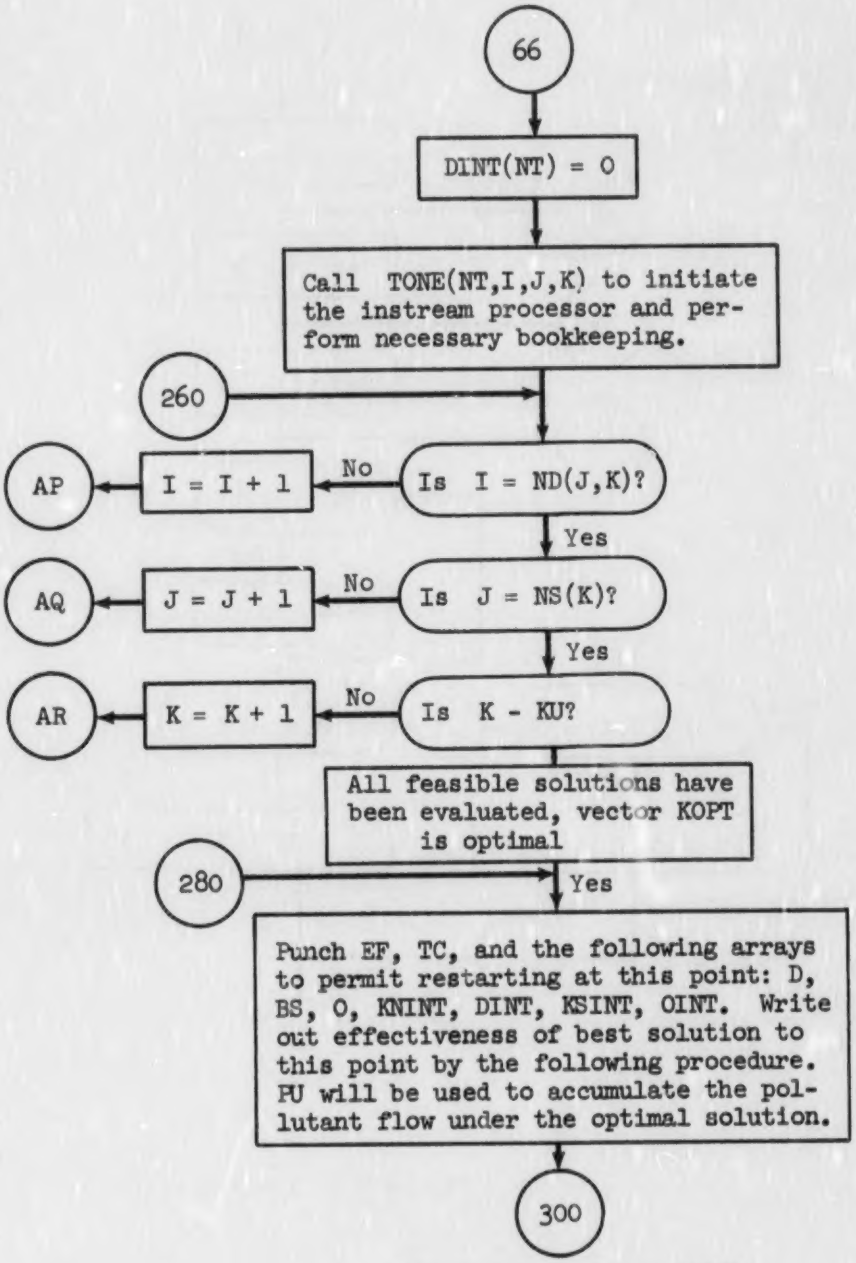

66
DINT(NT) = 0
Call TONE(NT,I,J,K) to initiate the instream processor and perform necessary bookkeeping.
260
Is I = ND(J,K)?
No
I = I + 1
AP
Yes
Is J = NS(K)?
No
J = J + 1
AQ
Yes
Is K - KU?
No
K = K + 1
AR
All feasible solutions have been evaluated, vector KOPT is optimal
280
Yes
Punch EF, TC, and the following arrays to permit restarting at this point: D, BS, O, KNINT, DINT, KSINT, OINT. Write out effectiveness of best solution to this point by the following procedure. PU will be used to accumulate the pollutant flow under the optimal solution.
300

Figure C.5 Program MAXEF

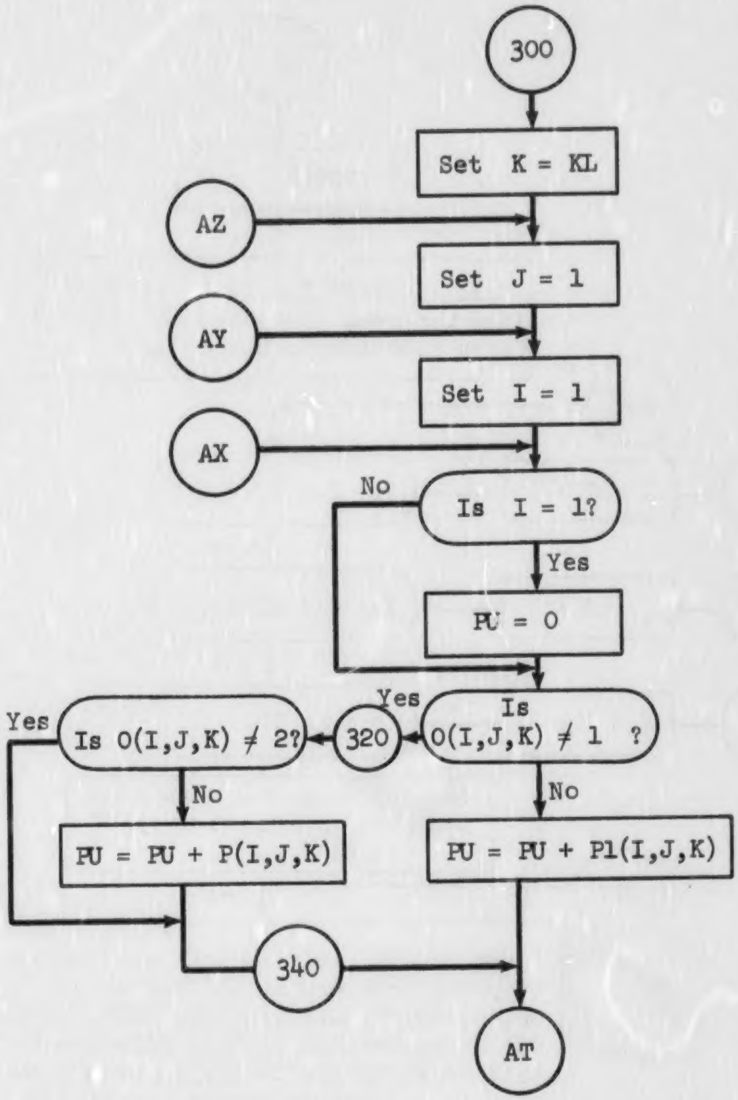

300
Set K = KL
AZ
Set J = 1
AY
Set I = 1
AX
No
Is I = 1?
Yes
PU = 0
Yes
Is O(I,J,K) ≠ 1 ?
320
Yes
Is O(I,J,K) ≠ 2?
No
PU = PU + P(I,J,K)
No
PU = PU + P1(I,J,K)
340
AT

Figure C.5 Program MAXEF

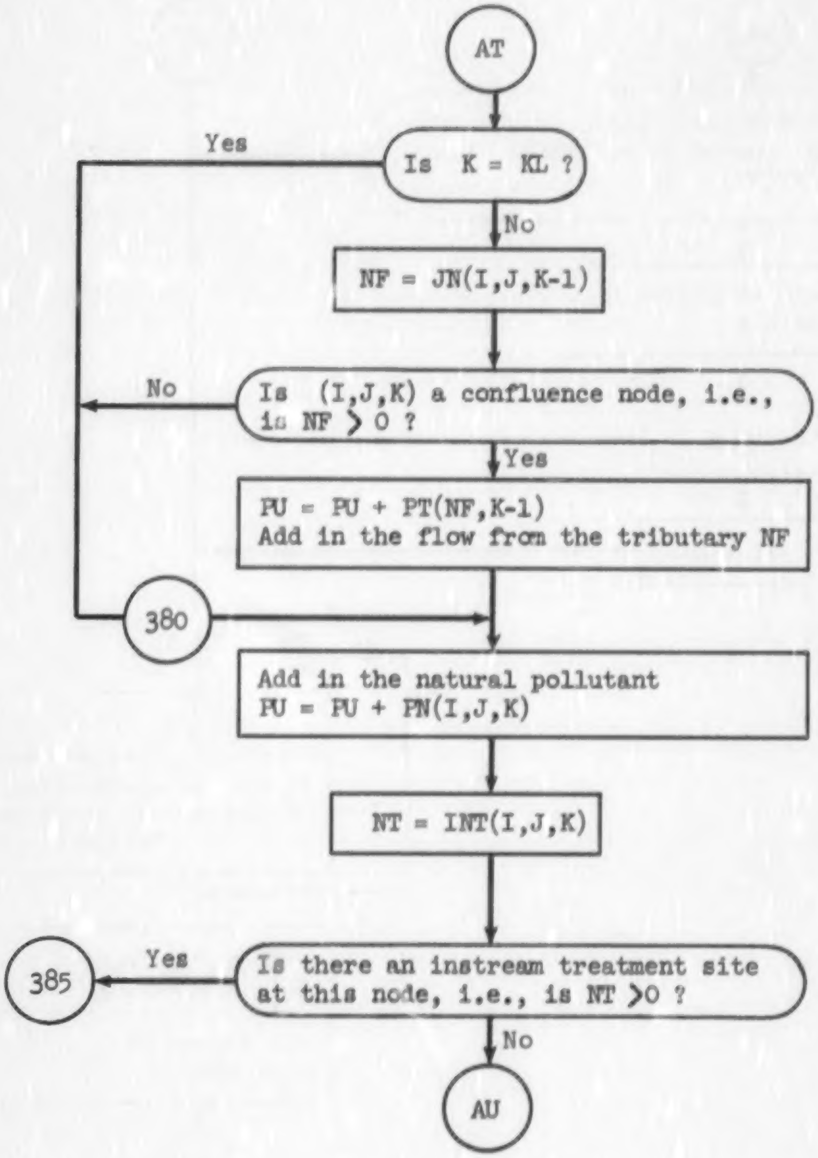

AT
Is K = KL ?
Yes
No
NF = JN(I,J,K-1)
Is (I,J,K) a confluence node, i.e., is NF > 0 ?
No
Yes
PU = PU + PT(NF,K-1)
Add in the flow from the tributary NF
380
Add in the natural pollutant
PU = PU + PN(I,J,K)
NT = INT(I,J,K)
Is there an instream treatment site at this node, i.e., is NT >0 ?
Yes
385
No
AU

Figure C.5 Program MAXEF

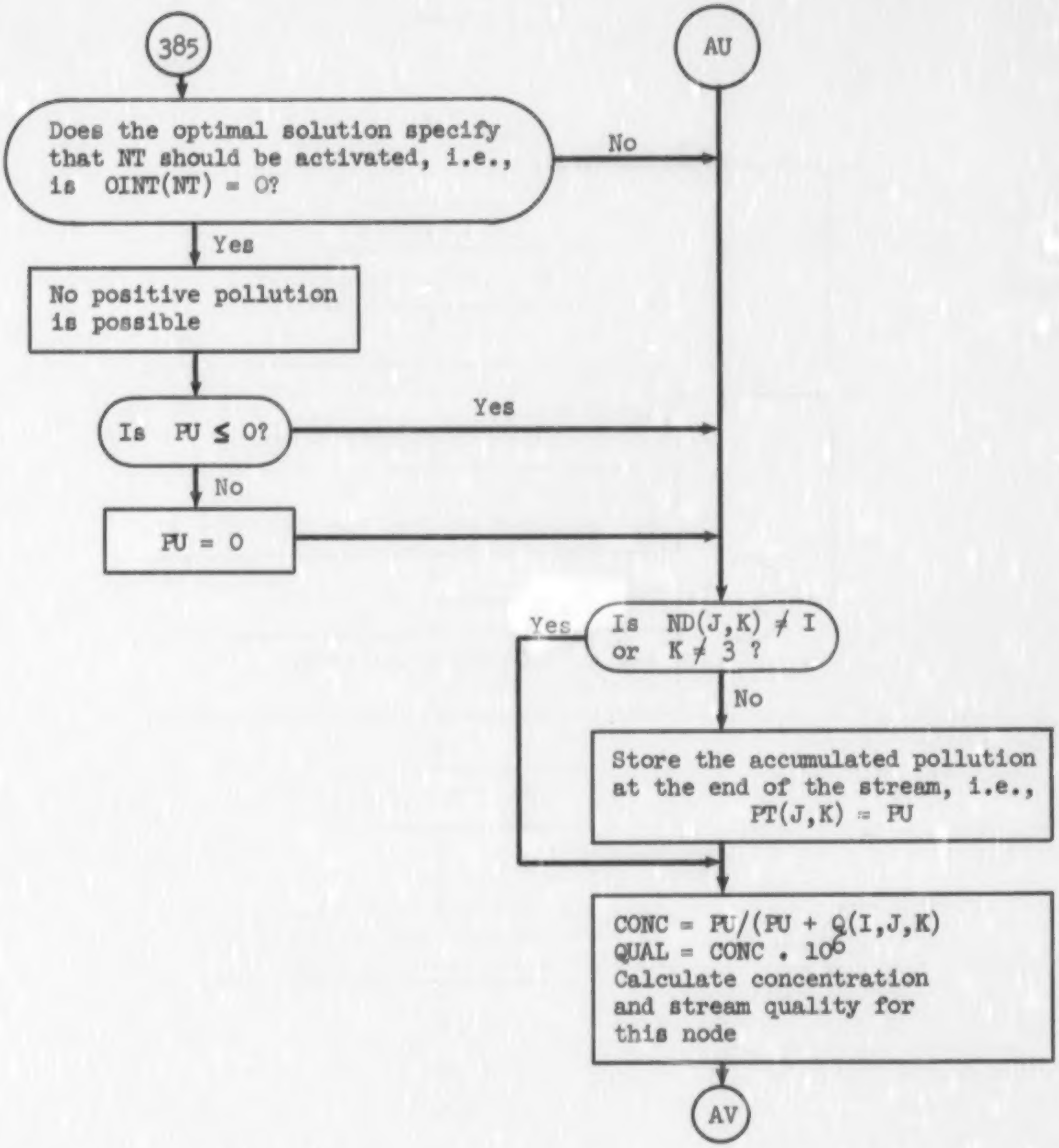

385
AU
Does the optimal solution specify that NT should be activated, i.e., is OINT(NT) = O?
No
Yes
No positive pollution is possible
Is PU ≤ O?
Yes
No
PU = 0
Yes
Is ND(J,K) ≠ I or K ≠ 3 ?
No
Store the accumulated pollution at the end of the stream, i.e., PT(J,K) = PU
CONC = PU/(PU + Q(I,J,K)
QUAL = CONC . 10^6
Calculate concentration and stream quality for this node
AV

Figure C.5 Program MAXEF

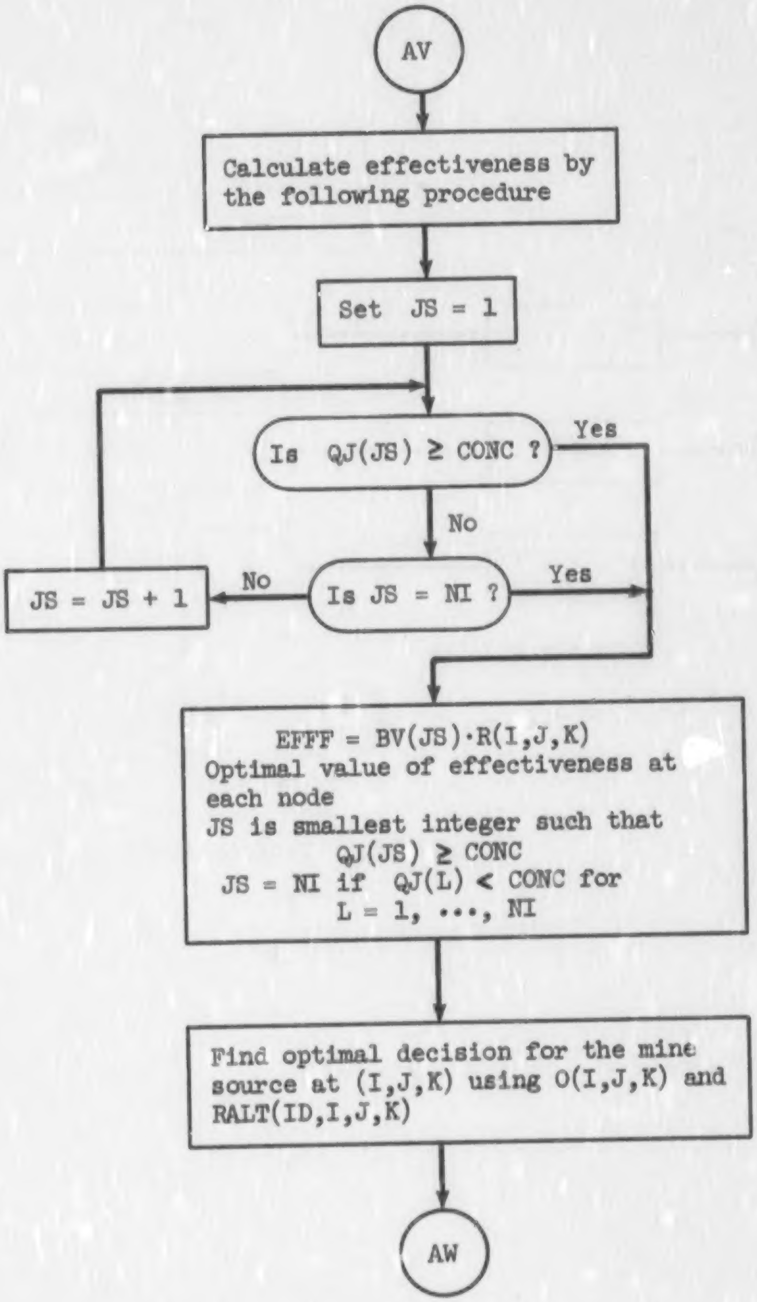

AV
Calculate effectiveness by the following procedure
Set JS = 1
Is QJ(JS) ≥ CONC ?
Yes
No
Is JS = NI ?
Yes
No
JS = JS + 1
EFFF = BV(JS)·R(I,J,K)
Optimal value of effectiveness at each node
JS is smallest integer such that QJ(JS) ≥ CONC
JS = NI if QJ(L) < CONC for L = 1, ..., NI
Find optimal decision for the mine source at (I,J,K) using O(I,J,K) and RALT(ID,I,J,K)
AW

Figure C.5 Program MAXEF

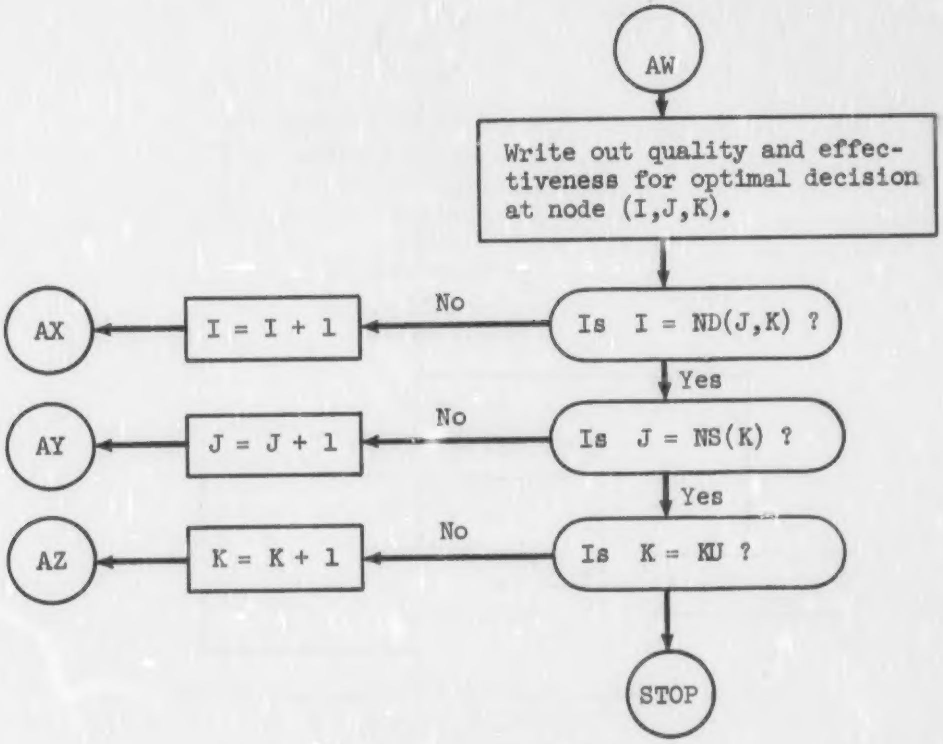

AW
Write out quality and effectiveness for optimal decision at node (I,J,K).
Is I = ND(J,K) ?
No
I = I + 1
AX
Yes
Is J = NS(K) ?
No
J = J + 1
AY
Yes
Is K = KU ?
No
K = K + 1
AZ
STOP

Figure C.5 Program MAXEF

Function CABAT(II,JJ,KK)

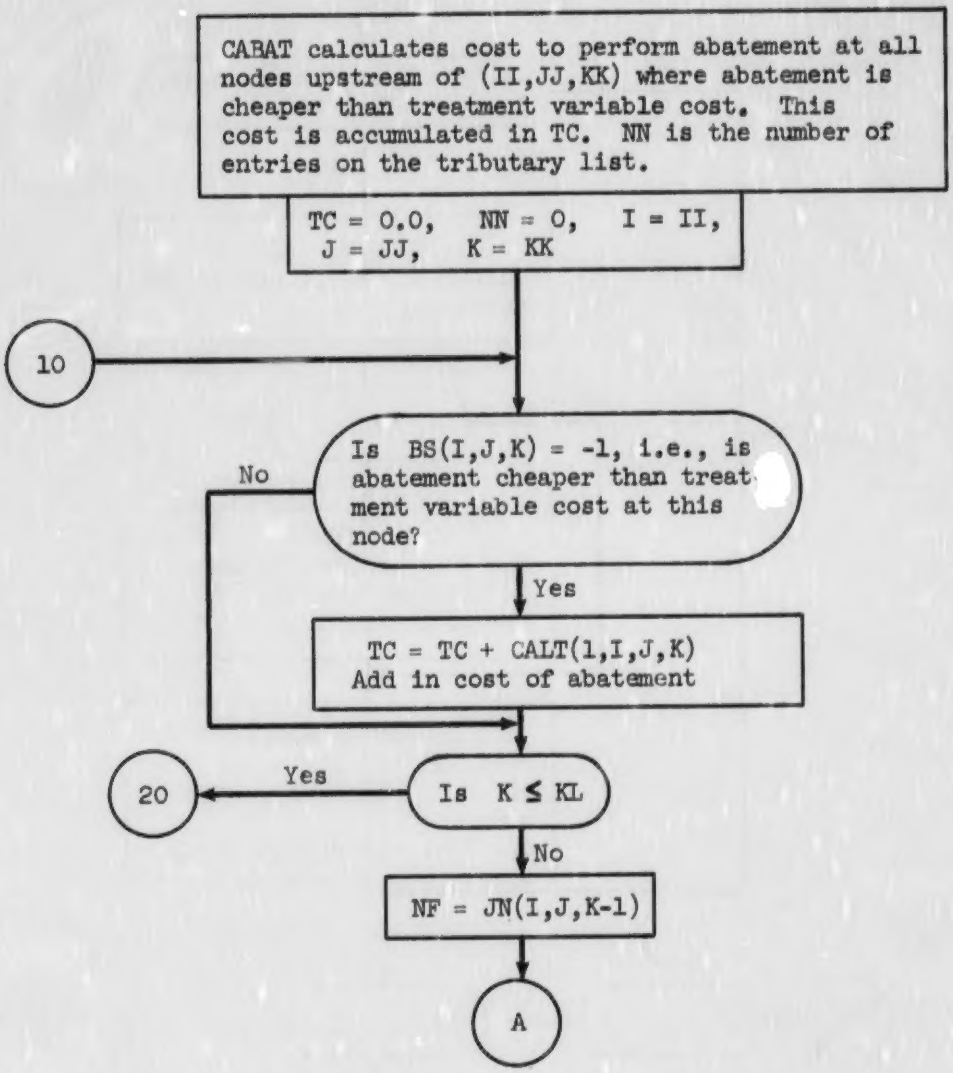

CABAT calculates cost to perform abatement at all nodes upstream of (II,JJ,KK) where abatement is cheaper than treatment variable cost. This cost is accumulated in TC. NN is the number of entries on the tributary list.
TC = 0.0, NN = 0, I = II, J = JJ, K = KK
10
Is BS(I,J,K) = -1, i.e., is abatement cheaper than treatment variable cost at this node?
No
Yes
TC = TC + CALT(1,I,J,K)
Add in cost of abatement
20
Yes
Is K ≤ KL
No
NF = JN(I,J,K-1)
A

Figure C.6 Function CABAT

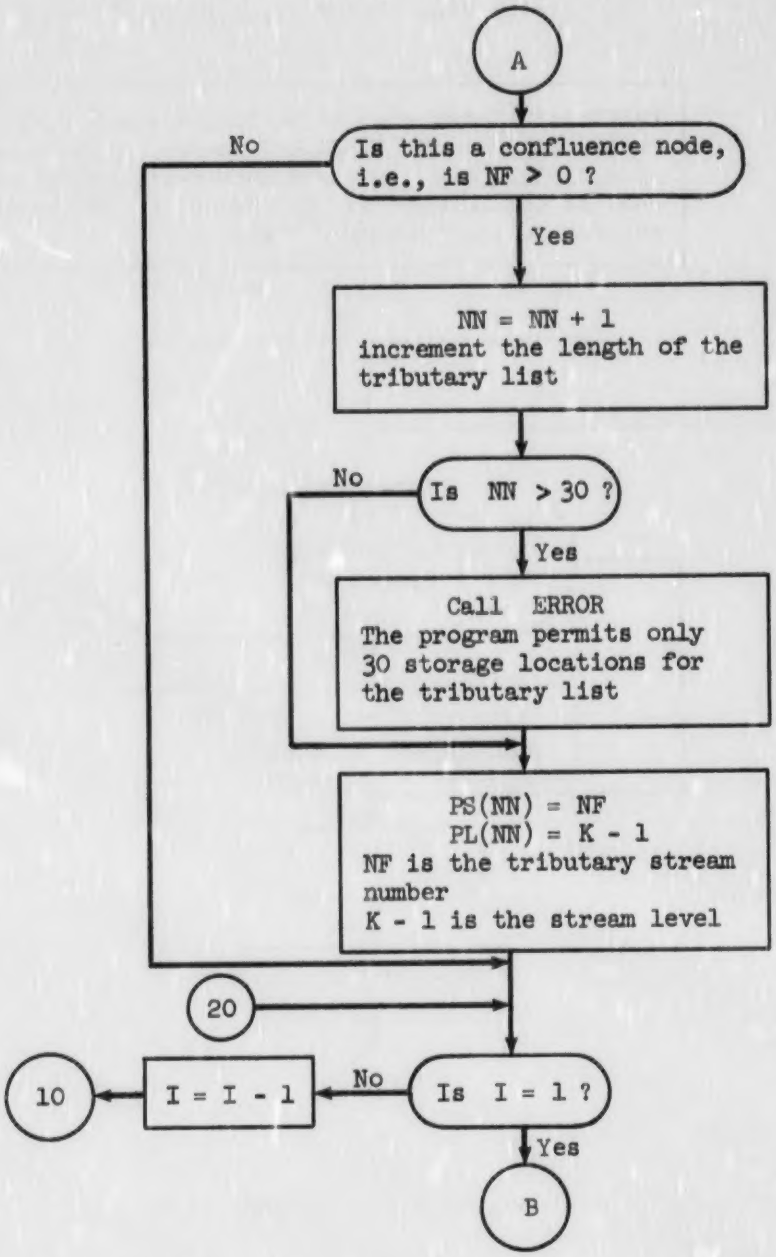

A
No
Is this a confluence node, i.e., is NF > 0 ?
Yes
NN = NN + 1
increment the length of the tributary list
No
Is NN > 30 ?
Yes
Call ERROR
The program permits only 30 storage locations for the tributary list
PS(NN) = NF
PL(NN) = K - 1
NF is the tributary stream number
K - 1 is the stream level
20
10
I = I - 1
No
Is I = 1 ?
Yes
B

Figure C.6 Function CABAT

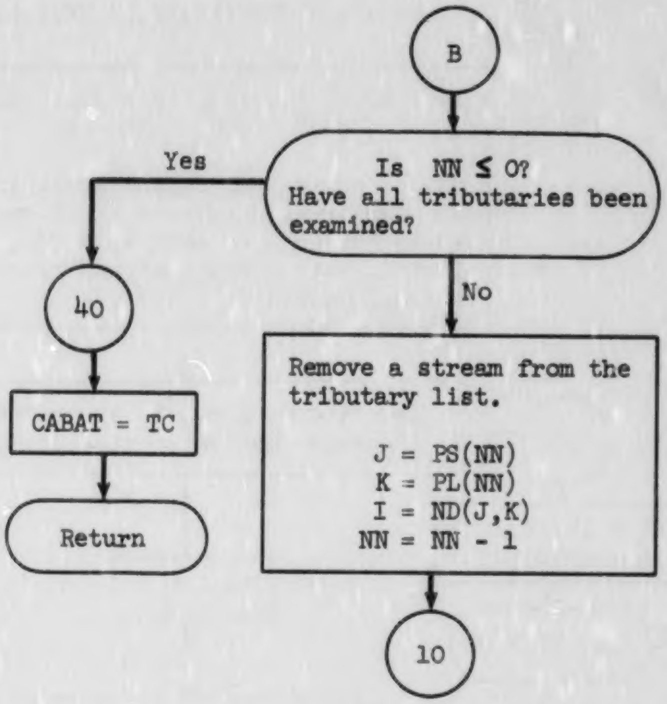
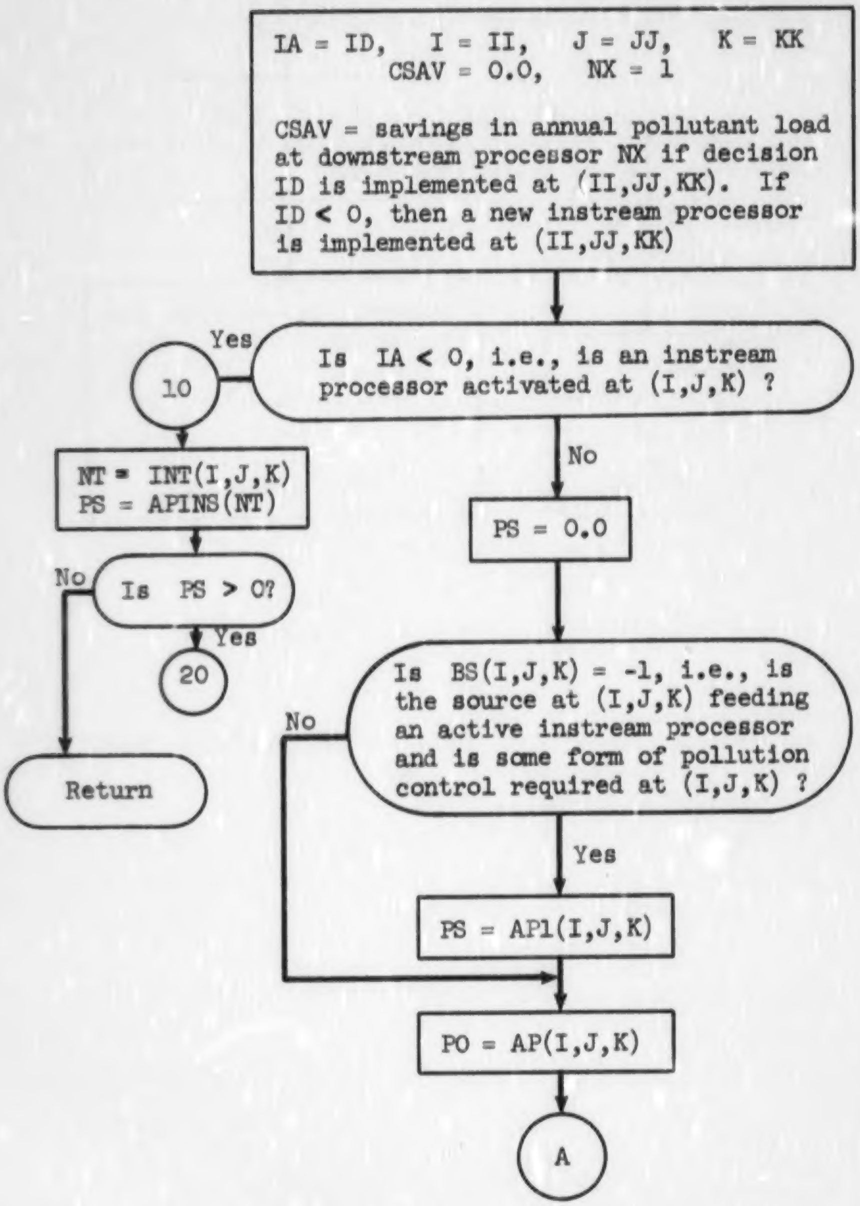

B
Yes
Is NN ≤ 0?
Have all tributaries been examined?
No
40
CABAT = TC
Return
Remove a stream from the tributary list.
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN - 1
10

Figure C.6 Function CABAT

Function CSAV(ID,II,JJ,KK,IS1,NX)



IA = ID, I = II, J = JJ, K = KK
CSAV = 0.0, NX = 1
CSAV = savings in annual pollutant load at downstream processor NX if decision ID is implemented at (II,JJ,KK). If ID < 0, then a new instream processor is implemented at (II,JJ,KK)
Is IA < 0, i.e., is an instream processor activated at (I,J,K) ?
Yes
10
NT = INT(I,J,K)
PS = APINS(NT)
Is PS > 0?
No
Yes
20
Return
No
PS = 0.0
Is BS(I,J,K) = -1, i.e., is the source at (I,J,K) feeding an active instream processor and is some form of pollution control required at (I,J,K) ?
No
Yes
PS = AP1(I,J,K)
PO = AP(I,J,K)
A

Figure C.7 Function CSAV

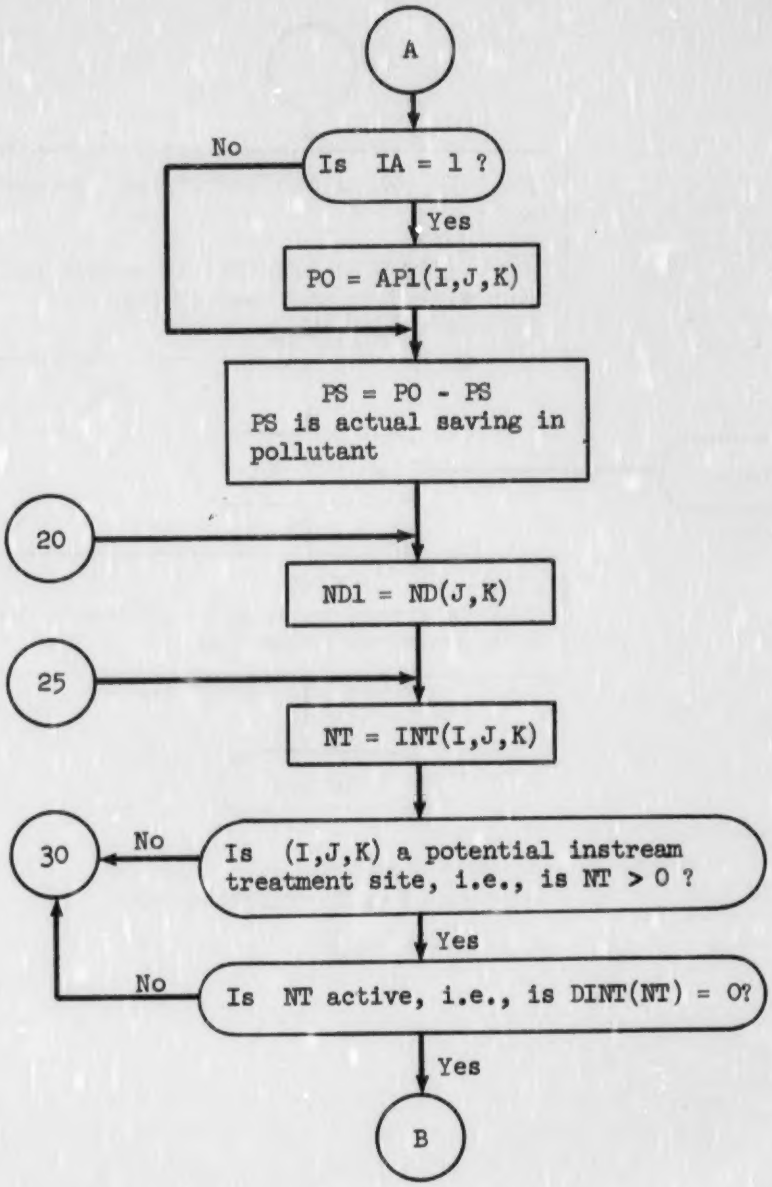
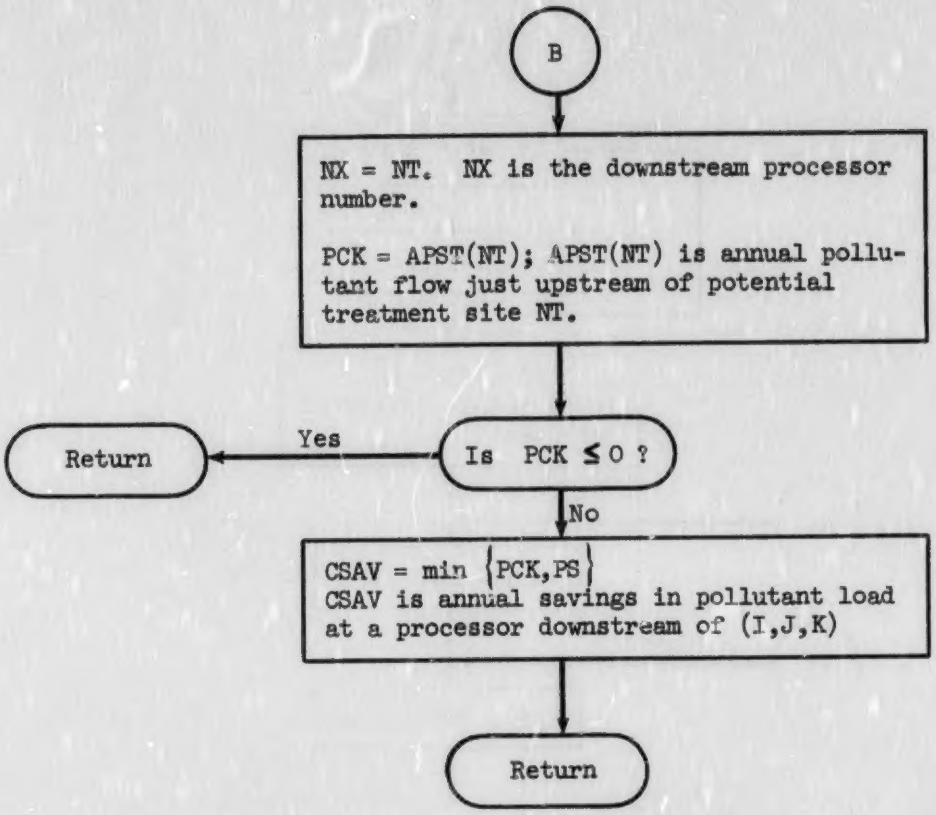

A
No
Is IA = 1 ?
Yes
PO = AP1(I,J,K)
PS = PO - PS
PS is actual saving in pollutant
20
ND1 = ND(J,K)
25
NT = INT(I,J,K)
30
No
Is (I,J,K) a potential instream treatment site, i.e., is NT > 0 ?
Yes
No
Is NT active, i.e., is DINT(NT) = 0?
Yes
B

Figure C.7 Function CSAV


B
Yes
Return
Is PCK ≤ 0 ?
No
Return

NX = NT. NX is the downstream processor number.

PCK = APST(NT); APST(NT) is annual pollutant flow just upstream of potential treatment site NT.

CSAV = min {PCK, PS}
CSAV is annual savings in pollutant load at a processor downstream of (I,J,K)

Figure C.7 Function CSAV

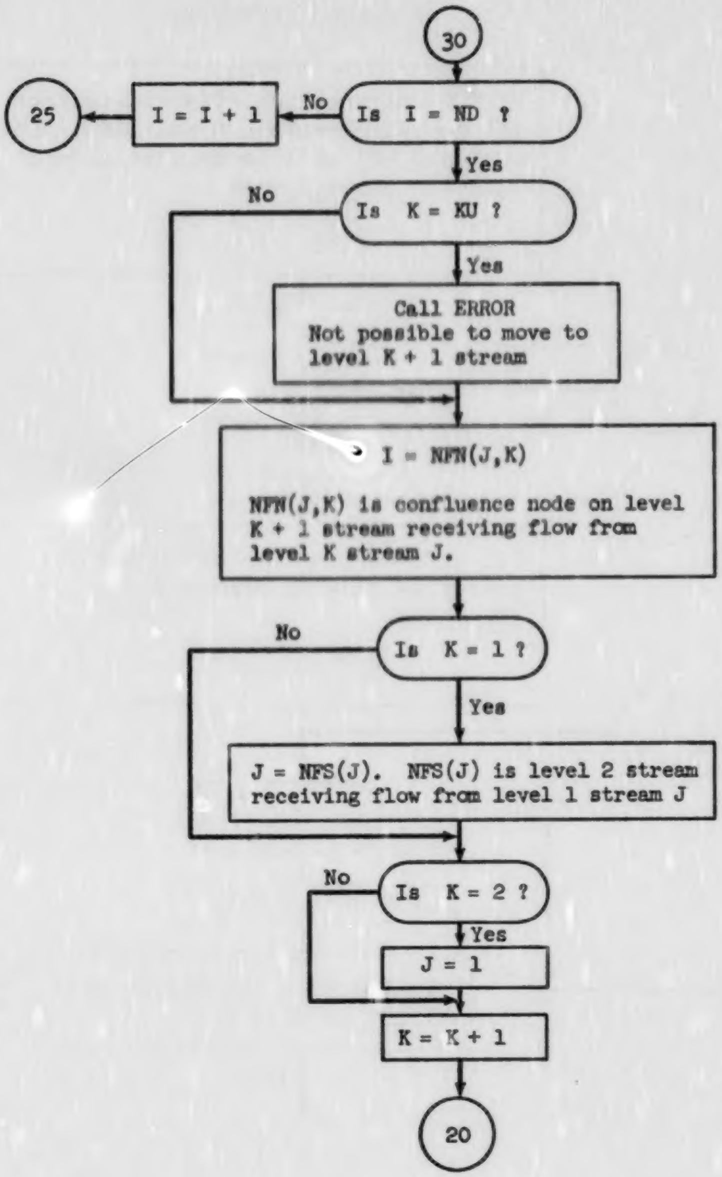

30
Is I = ND ?
No
I = I + 1
25
Yes
Is K = KU ?
No
Yes
Call ERROR
Not possible to move to level K + 1 stream
I = NFN(J,K)
NFN(J,K) is confluence node on level K + 1 stream receiving flow from level K stream J.
Is K = 1 ?
No
Yes
J = NFS(J). NFS(J) is level 2 stream receiving flow from level 1 stream J
Is K = 2 ?
No
Yes
J = 1
K = K + 1
20

Figure C.7 Function CSAV

Function EFFECT(X)

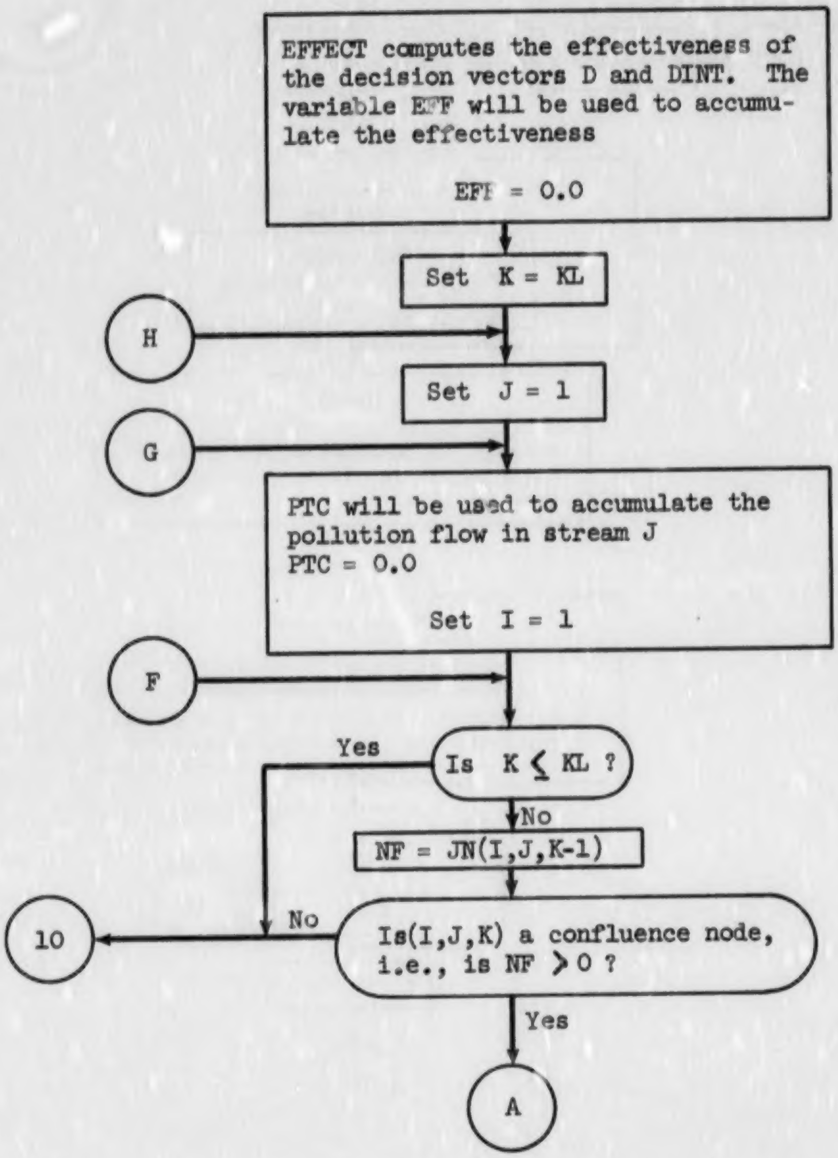

EFFECT computes the effectiveness of the decision vectors D and DINT. The variable EFF will be used to accumulate the effectiveness
EFF = 0.0
Set K = KL
H
Set J = 1
G
PTC will be used to accumulate the pollution flow in stream J
PTC = 0.0
Set I = 1
F
Yes
Is K ≤ KL ?
No
NF = JN(I,J,K-1)
10
No
Is(I,J,K) a confluence node, i.e., is NF > 0 ?
Yes
A

Figure C.8 Function EFFECT

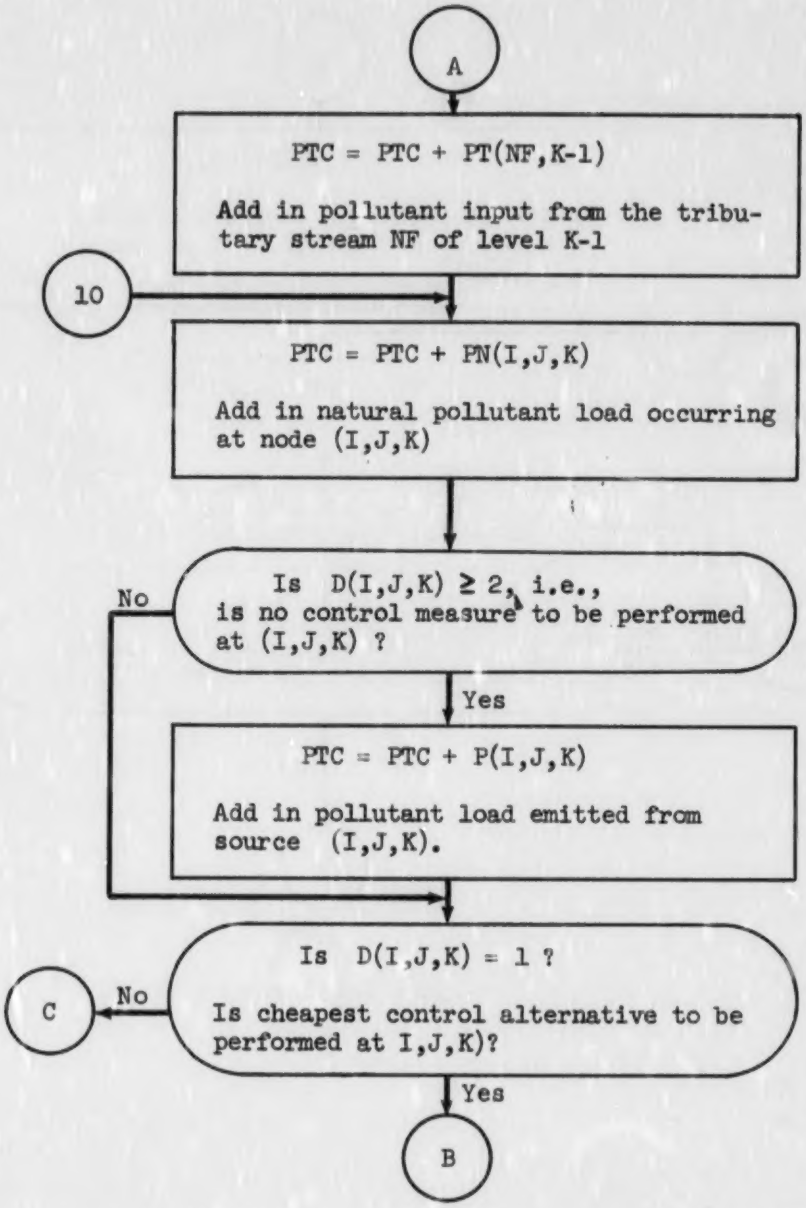

A
PTC = PTC + PT(NF,K-1)
Add in pollutant input from the tributary stream NF of level K-1
10
PTC = PTC + PN(I,J,K)
Add in natural pollutant load occurring at node (I,J,K)
Is D(I,J,K) ≥ 2, i.e., is no control measure to be performed at (I,J,K) ?
No
Yes
PTC = PTC + P(I,J,K)
Add in pollutant load emitted from source (I,J,K).
Is D(I,J,K) = 1 ?
Is cheapest control alternative to be performed at I,J,K)?
No
C
Yes
B

Figure C.8 Function EFFECT

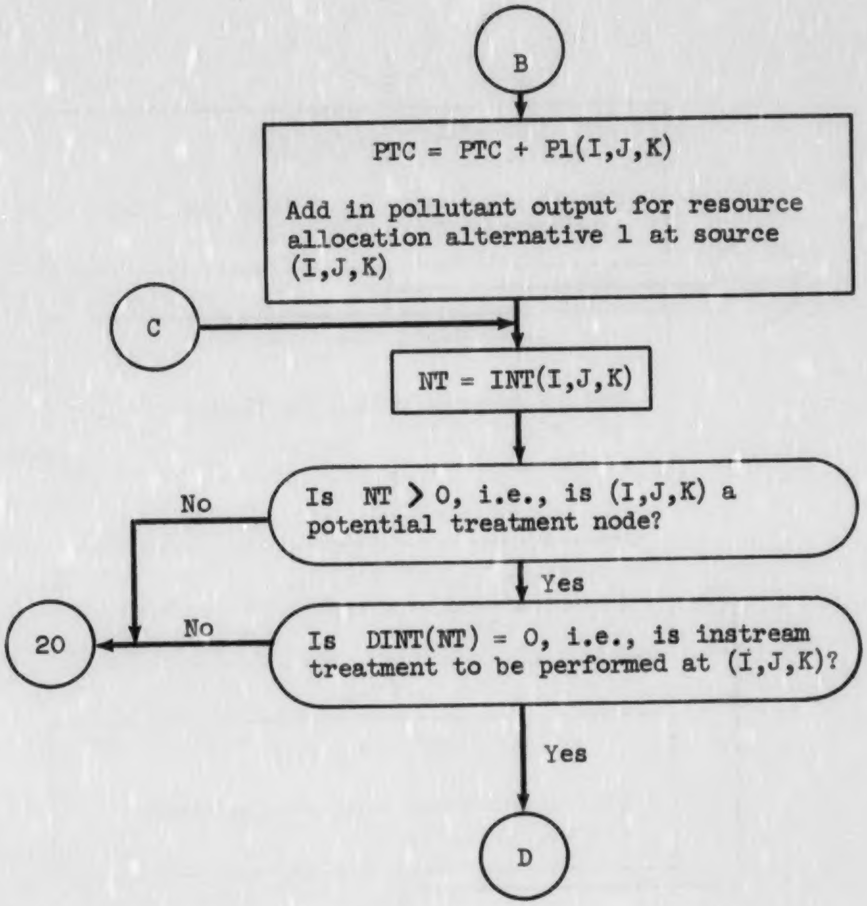
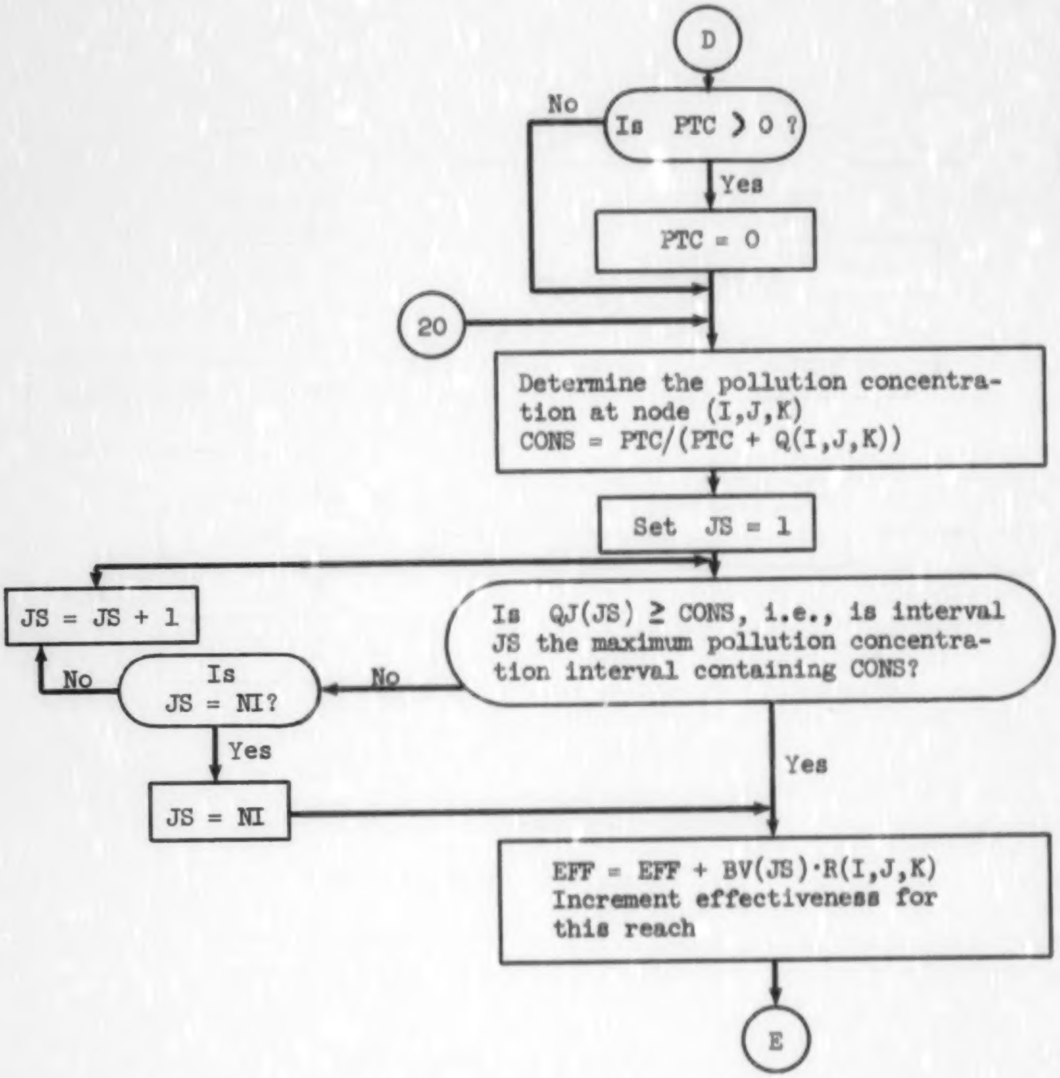

B
PTC = PTC + P1(I,J,K)
Add in pollutant output for resource allocation alternative 1 at source (I,J,K)
C
NT = INT(I,J,K)
No
Is NT > 0, i.e., is (I,J,K) a potential treatment node?
Yes
20
No
Is DINT(NT) = 0, i.e., is instream treatment to be performed at (I,J,K)?
Yes
D

Figure C.8 Function EFFECT



D
Is PTC > 0 ?
No
Yes
PTC = 0
20
Determine the pollution concentration at node (I,J,K)
CONS = PTC/(PTC + Q(I,J,K))
Set JS = 1
JS = JS + 1
Is QJ(JS) ≥ CONS, i.e., is interval JS the maximum pollution concentration interval containing CONS?
No
Is JS = NI?
No
Yes
JS = NI
Yes
EFF = EFF + BV(JS)·R(I,J,K)
Increment effectiveness for this reach
E

Figure C.8 Function EFFECT

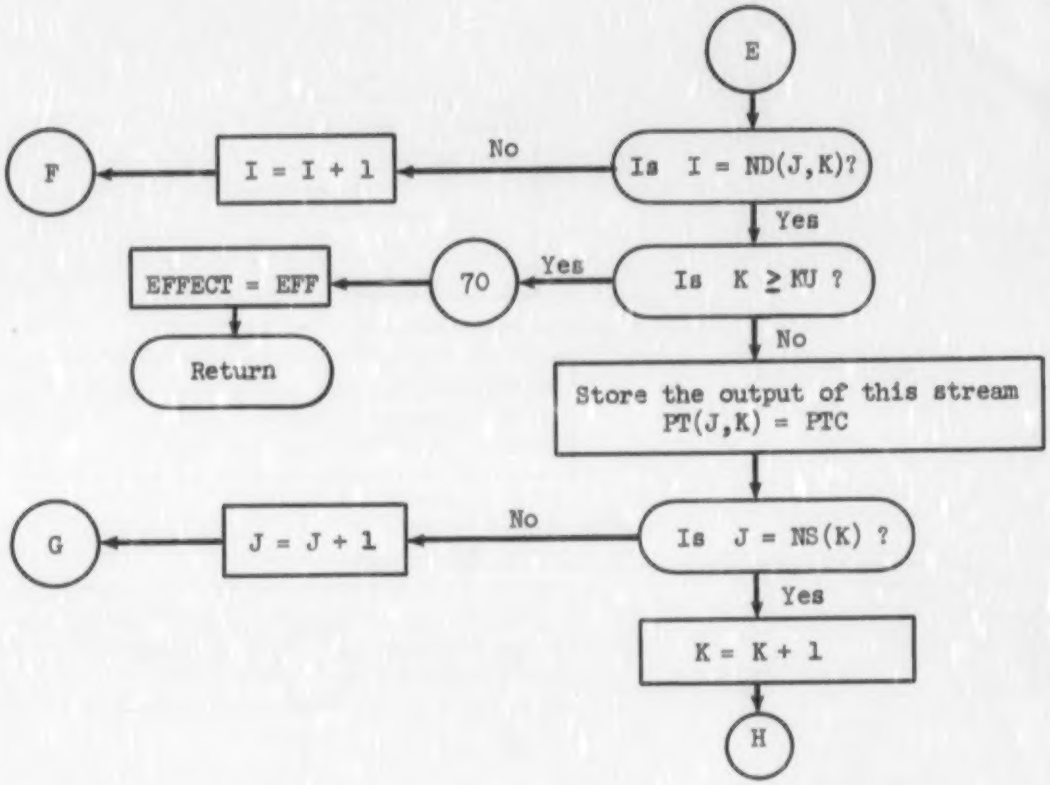

E
Is I = ND(J,K)?
No
I = I + 1
F
Yes
Is K ≥ KU ?
Yes
70
EFFECT = EFF
Return
No
Store the output of this stream
PT(J,K) = PTC
Is J = NS(K) ?
No
J = J + 1
G
Yes
K = K + 1
H

Figure C.8 Function EFFECT

Subroutine ERROR

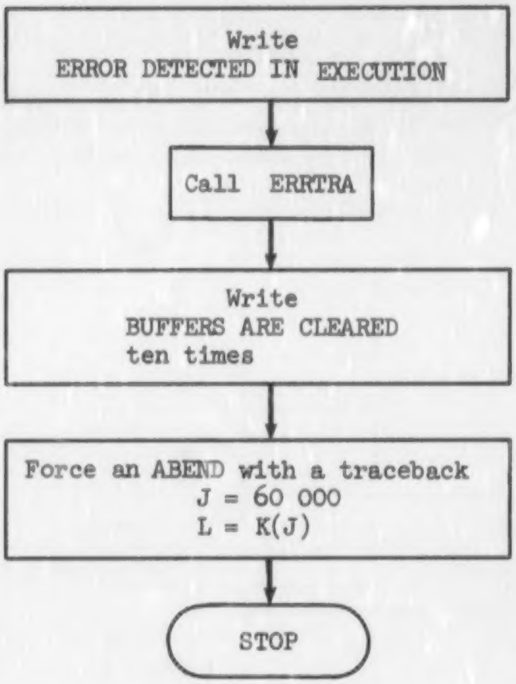
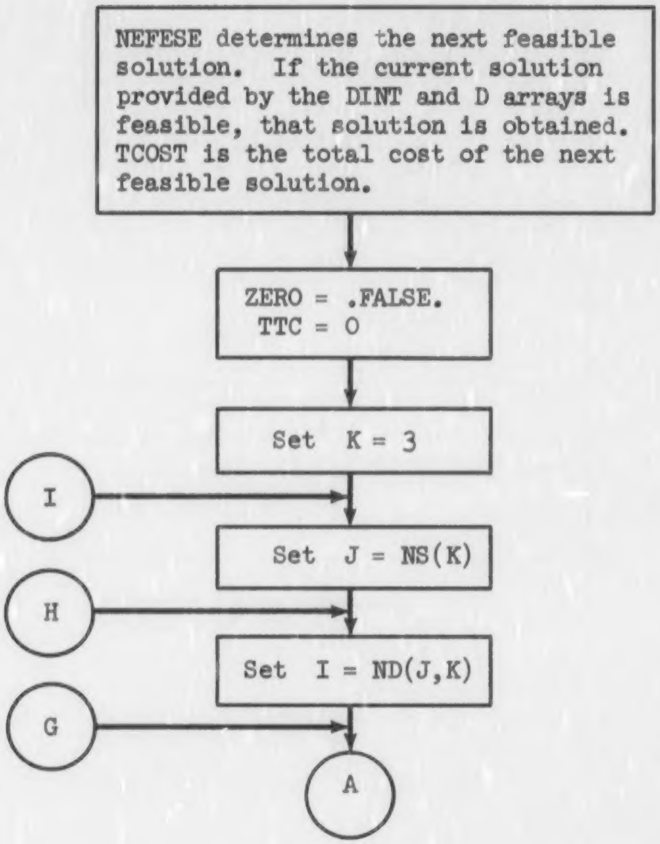

Write
ERROR DETECTED IN EXECUTION
Call ERRTRA
Write
BUFFERS ARE CLEARED
ten times
Force an ABEND with a traceback
J = 60 000
L = K(J)
STOP

Figure C.9 Subroutine ERROR

Subroutine NEFESE(TCOST)



NEFESE determines the next feasible solution. If the current solution provided by the DINT and D arrays is feasible, that solution is obtained. TCOST is the total cost of the next feasible solution.

ZERO = .FALSE.
TTC = 0
Set K = 3
I
Set J = NS(K)
H
Set I = ND(J,K)
G
A

Figure C.10 Subroutine NEFESE

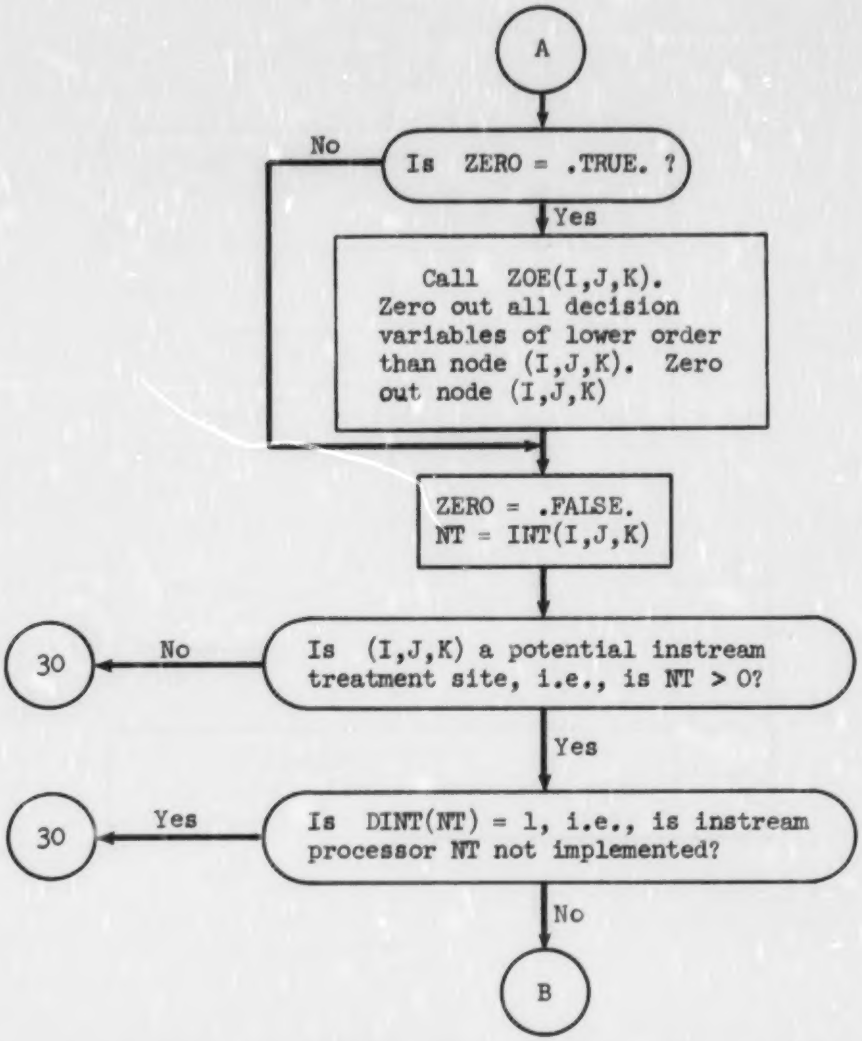

A
Is ZERO = .TRUE. ?
No
Yes
Call ZOE(I,J,K).
Zero out all decision variables of lower order than node (I,J,K). Zero out node (I,J,K)
ZERO = .FALSE.
NT = INT(I,J,K)
Is (I,J,K) a potential instream treatment site, i.e., is NT > 0?
No
30
Yes
Is DINT(NT) = 1, i.e., is instream processor NT not implemented?
Yes
30
No
B

Figure C.10 Subroutine NEFESE

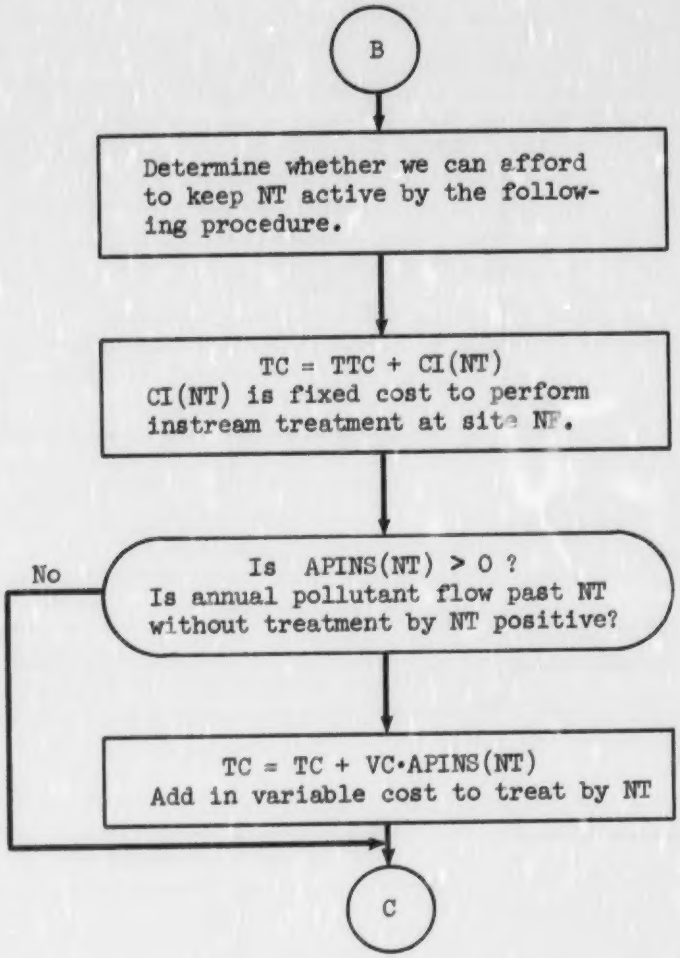
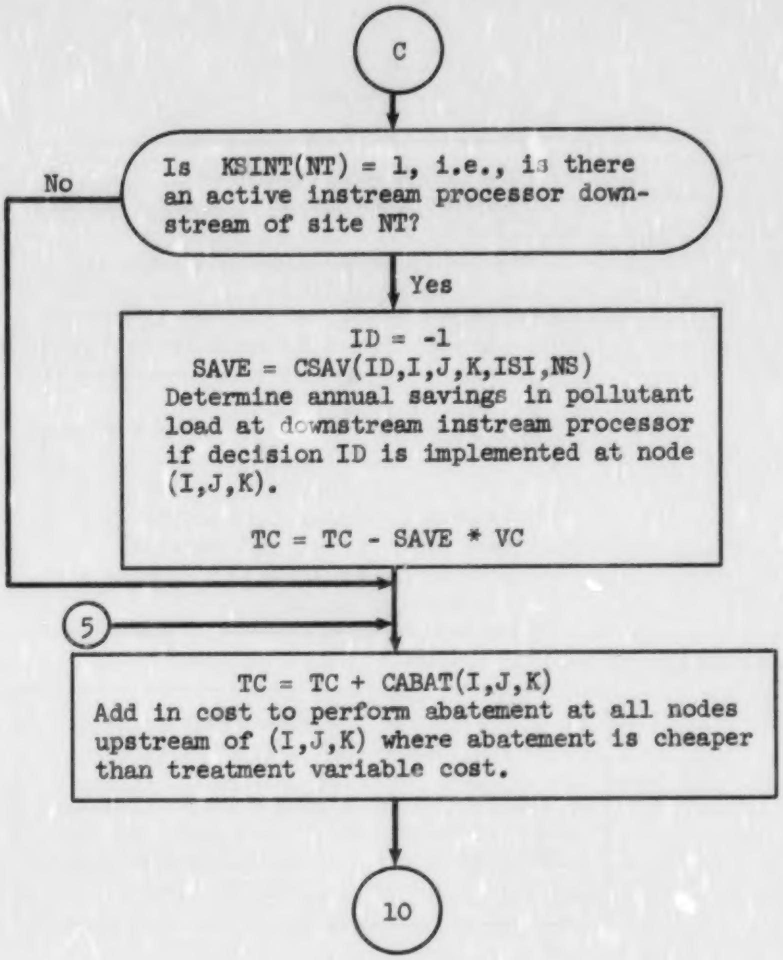

B
Determine whether we can afford to keep NT active by the following procedure.
TC = TTC + CI(NT)
CI(NT) is fixed cost to perform instream treatment at site NF.
No
Is APINS(NT) > 0 ?
Is annual pollutant flow past NT without treatment by NT positive?
TC = TC + VC•APINS(NT)
Add in variable cost to treat by NT
C

Figure C.10 Subroutine NEFESE



Is KSINT(NT) = 1, i.e., is there an active instream processor downstream of site NT?

ID = -1
SAVE = CSAV(ID,I,J,K,ISI,NS)
Determine annual savings in pollutant load at downstream instream processor if decision ID is implemented at node (I,J,K).

TC = TC - SAVE * VC

TC = TC + CABAT(I,J,K)
Add in cost to perform abatement at all nodes upstream of (I,J,K) where abatement is cheaper than treatment variable cost.

Figure C.10 Subroutine NFFESE

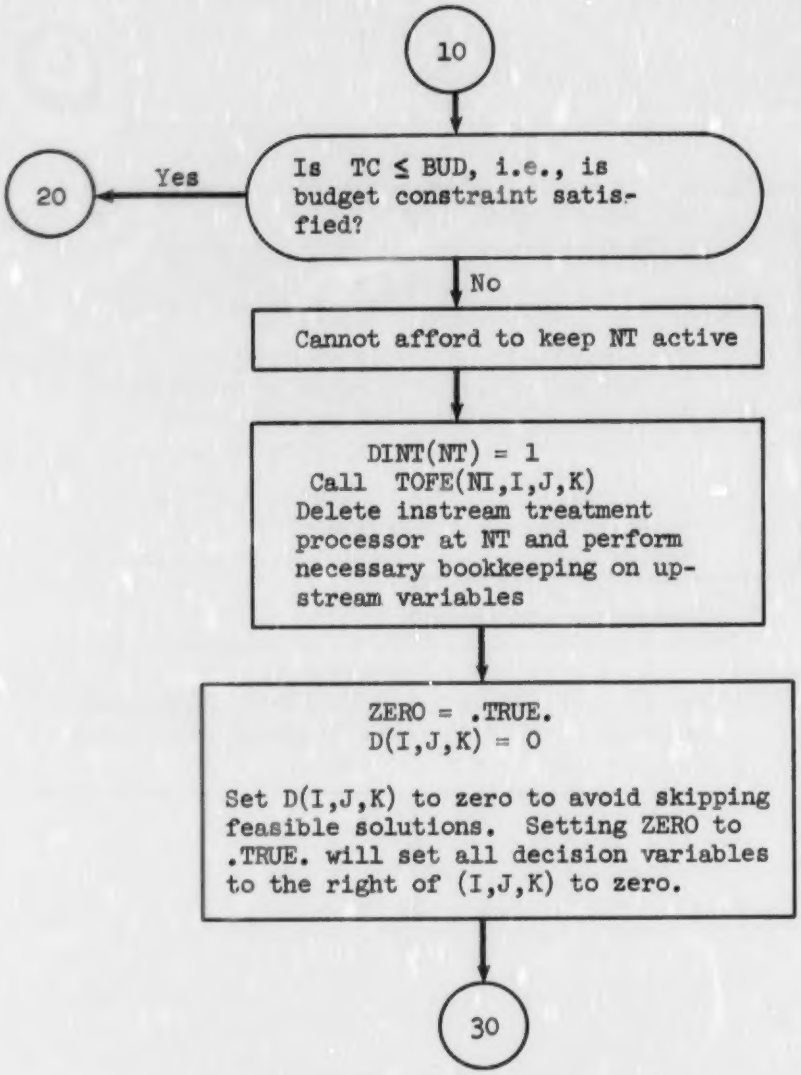

10
Is TC ≤ BUD, i.e., is budget constraint satisfied?
Yes
20
No
Cannot afford to keep NT active
DINT(NT) = 1
Call TOFE(NI,I,J,K)
Delete instream treatment processor at NT and perform necessary bookkeeping on upstream variables
ZERO = .TRUE.
D(I,J,K) = 0
Set D(I,J,K) to zero to avoid skipping feasible solutions. Setting ZERO to .TRUE. will set all decision variables to the right of (I,J,K) to zero.
30

Figure C.10 Subroutine NEFESE

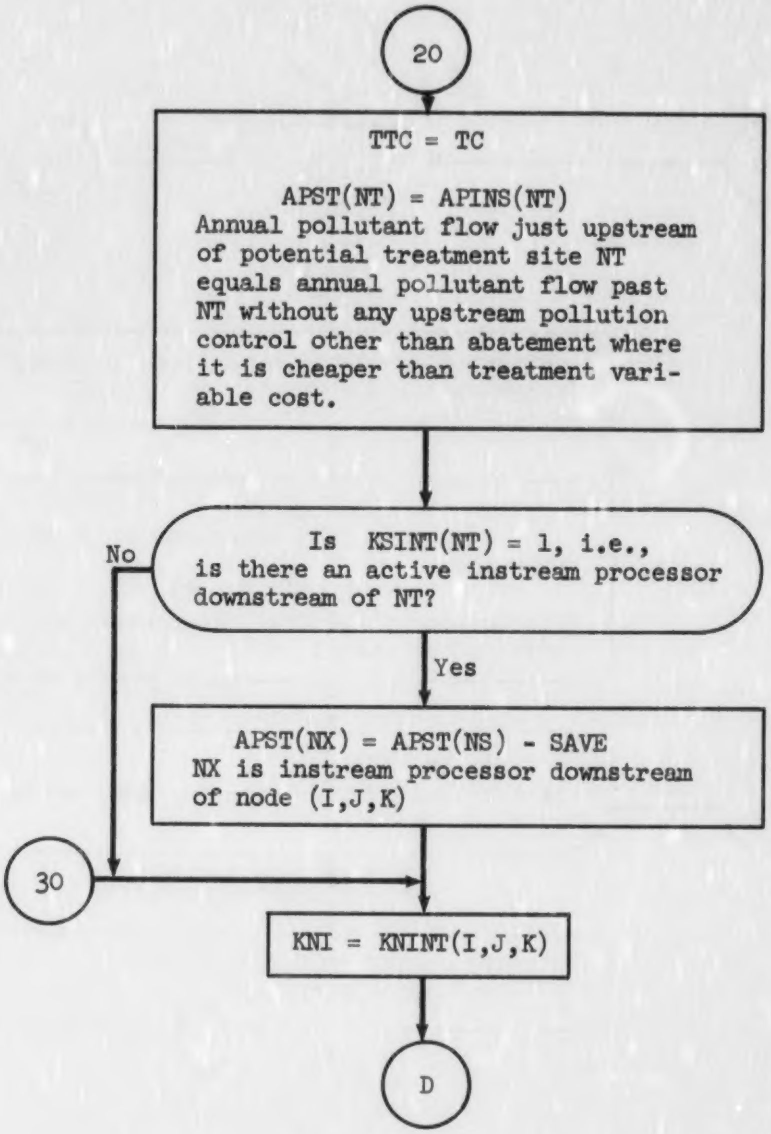

20
TTC = TC
APST(NT) = APINS(NT)
Annual pollutant flow just upstream of potential treatment site NT equals annual pollutant flow past NT without any upstream pollution control other than abatement where it is cheaper than treatment variable cost.
Is KSINT(NT) = 1, i.e., is there an active instream processor downstream of NT?
No
Yes
APST(NX) = APST(NS) - SAVE
NX is instream processor downstream of node (I,J,K)
30
KNI = KNINT(I,J,K)
D

Figure C.10 Subroutine NEFESE

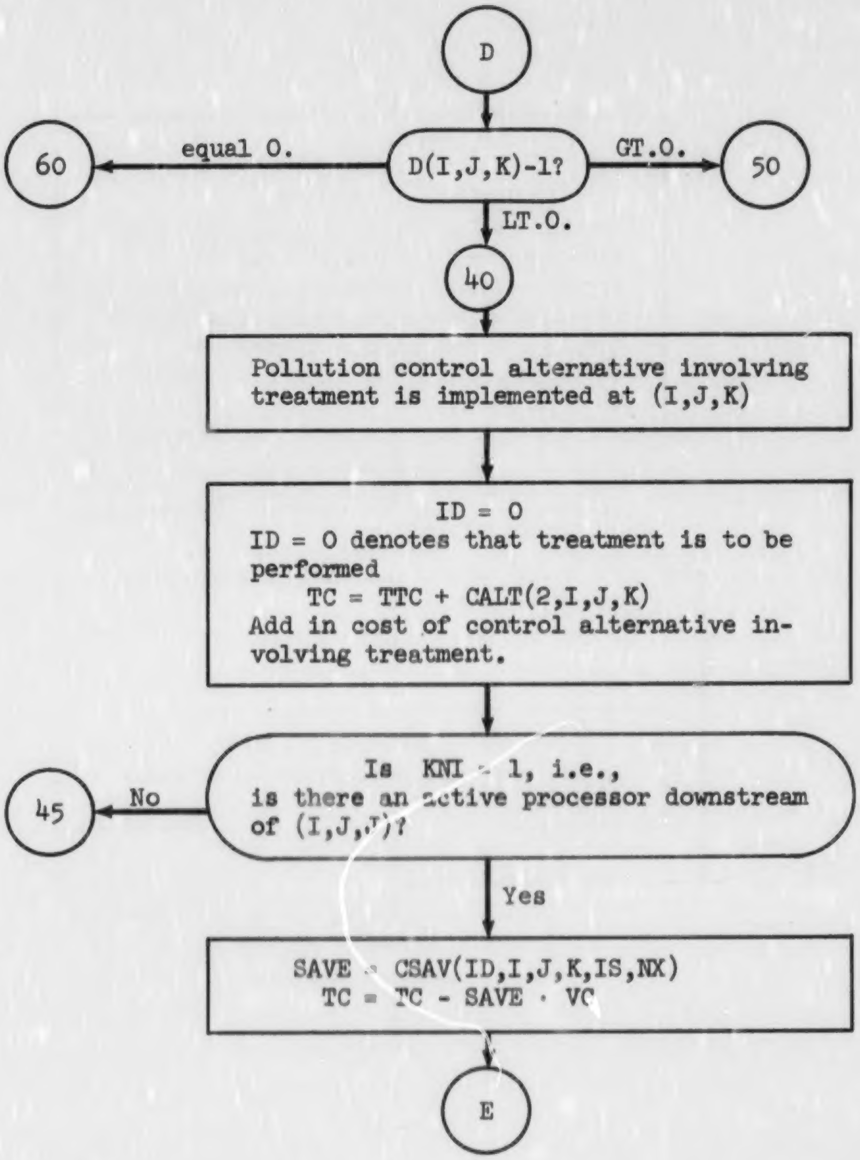

D
60
equal 0.
D(I,J,K)-1?
GT.0.
50
LT.0.
40
Pollution control alternative involving treatment is implemented at (I,J,K)
ID = 0
ID = 0 denotes that treatment is to be performed
TC = TTC + CALT(2,I,J,K)
Add in cost of control alternative involving treatment.
Is KNI = 1, i.e., is there an active processor downstream of (I,J,K)?
45
No
Yes
SAVE = CSAV(ID,I,J,K,IS,NX)
TC = TC - SAVE · VC
E

Figure C.10 Subroutine NEFESE

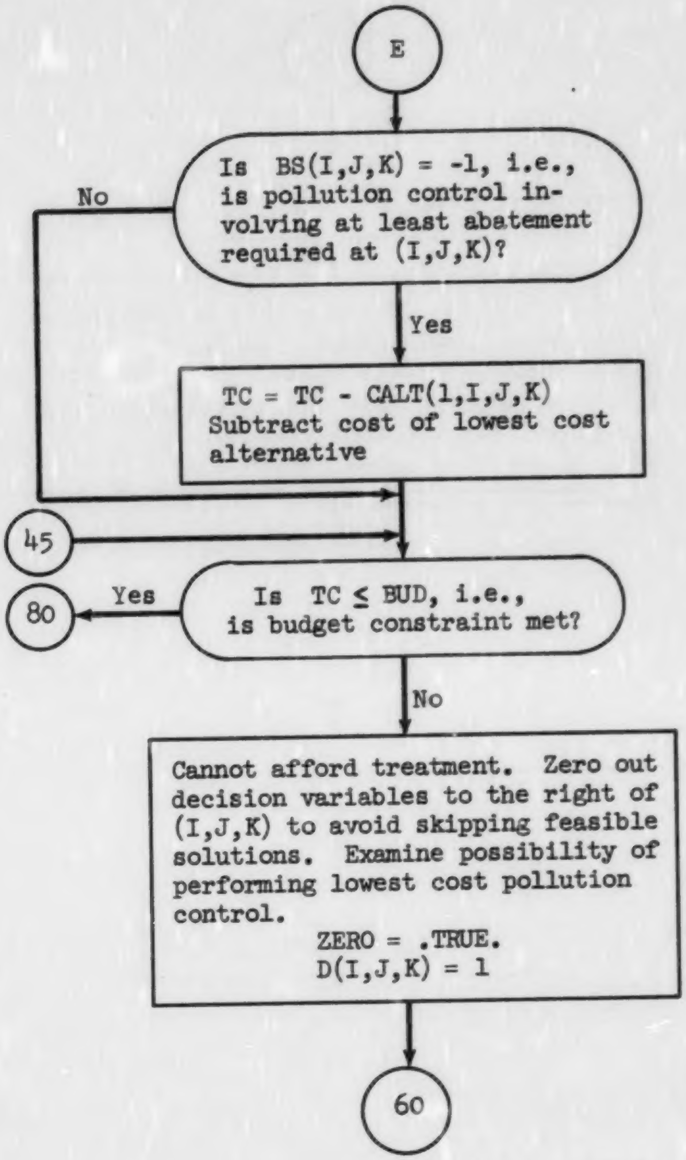

E
Is BS(I,J,K) = -1, i.e., is pollution control involving at least abatement required at (I,J,K)?
No
Yes
TC = TC - CALT(1,I,J,K) Subtract cost of lowest cost alternative
45
Is TC ≤ BUD, i.e., is budget constraint met?
Yes
80
No
Cannot afford treatment. Zero out decision variables to the right of (I,J,K) to avoid skipping feasible solutions. Examine possibility of performing lowest cost pollution control.
ZERO = .TRUE.
D(I,J,K) = 1
60

Figure C.10 Subroutine NEFESE

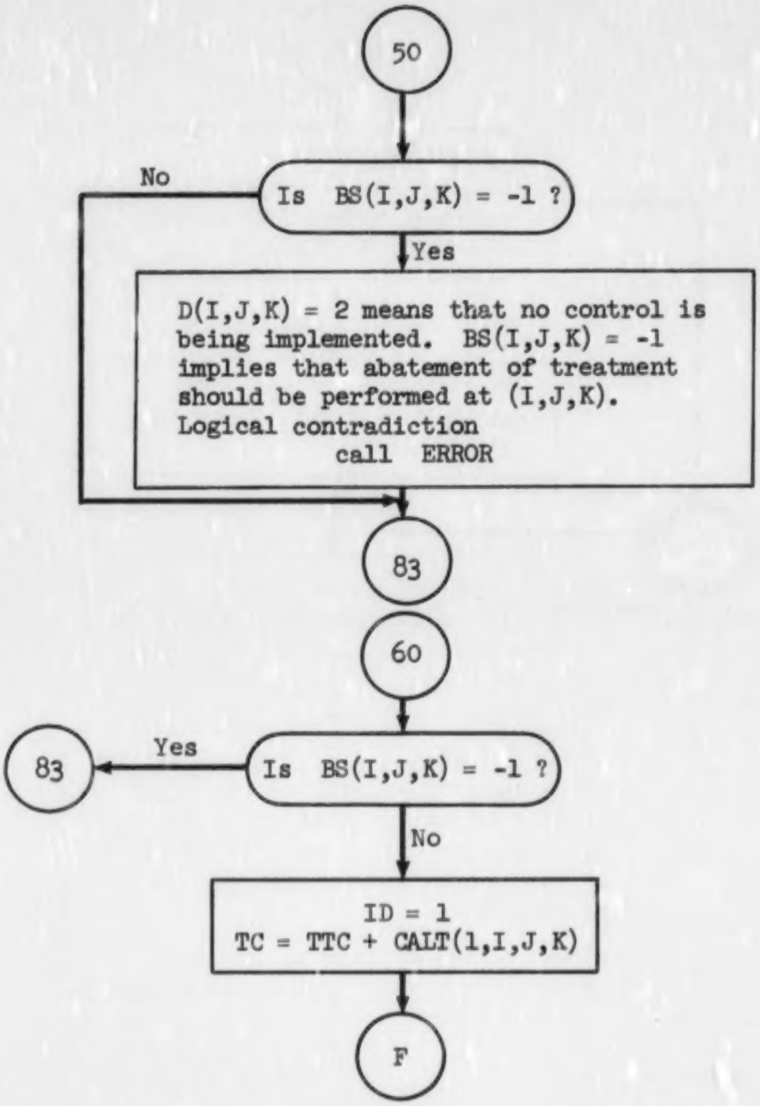

50
Is BS(I,J,K) = -1 ?
No
Yes
D(I,J,K) = 2 means that no control is being implemented. BS(I,J,K) = -1 implies that abatement of treatment should be performed at (I,J,K).
Logical contradiction
call ERROR
83
60
Yes
83
Is BS(I,J,K) = -1 ?
No
ID = 1
TC = TTC + CALT(1,I,J,K)
F

Figure C.10 Subroutine NEFESE

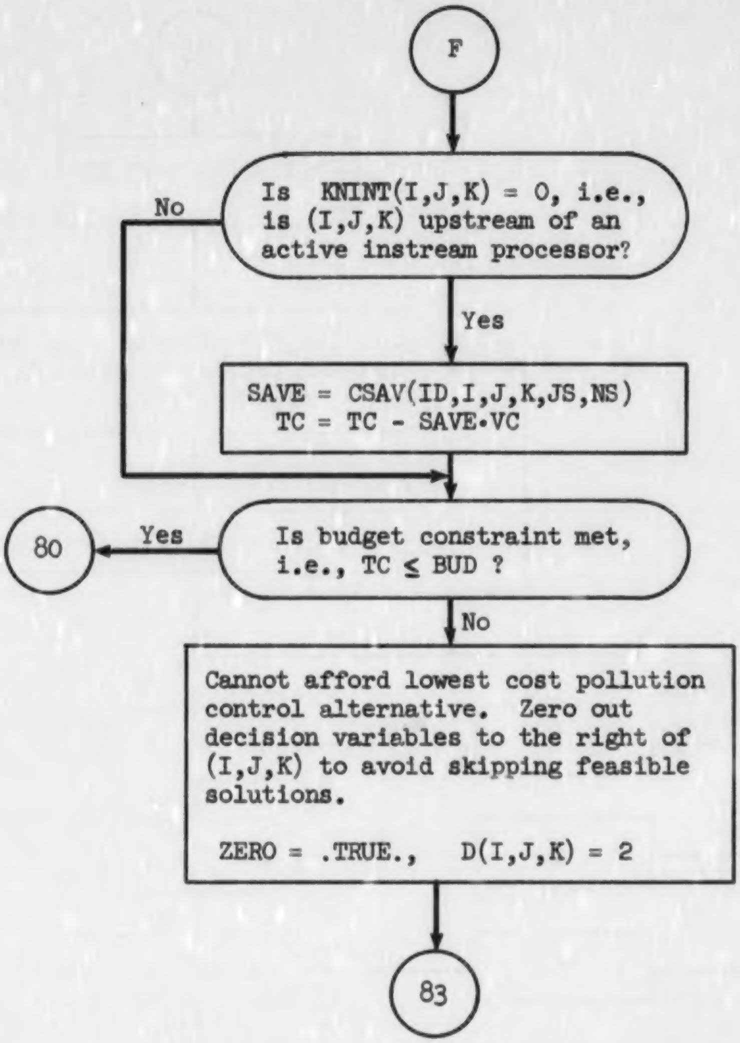

F
Is KNINT(I,J,K) = 0, i.e., is (I,J,K) upstream of an active instream processor?
No
Yes
SAVE = CSAV(ID,I,J,K,JS,NS)
TC = TC - SAVE•VC
80
Yes
Is budget constraint met, i.e., TC ≤ BUD ?
No
Cannot afford lowest cost pollution control alternative. Zero out decision variables to the right of (I,J,K) to avoid skipping feasible solutions.
ZERO = .TRUE., D(I,J,K) = 2
83

Figure C.10 Subroutine NEFESE

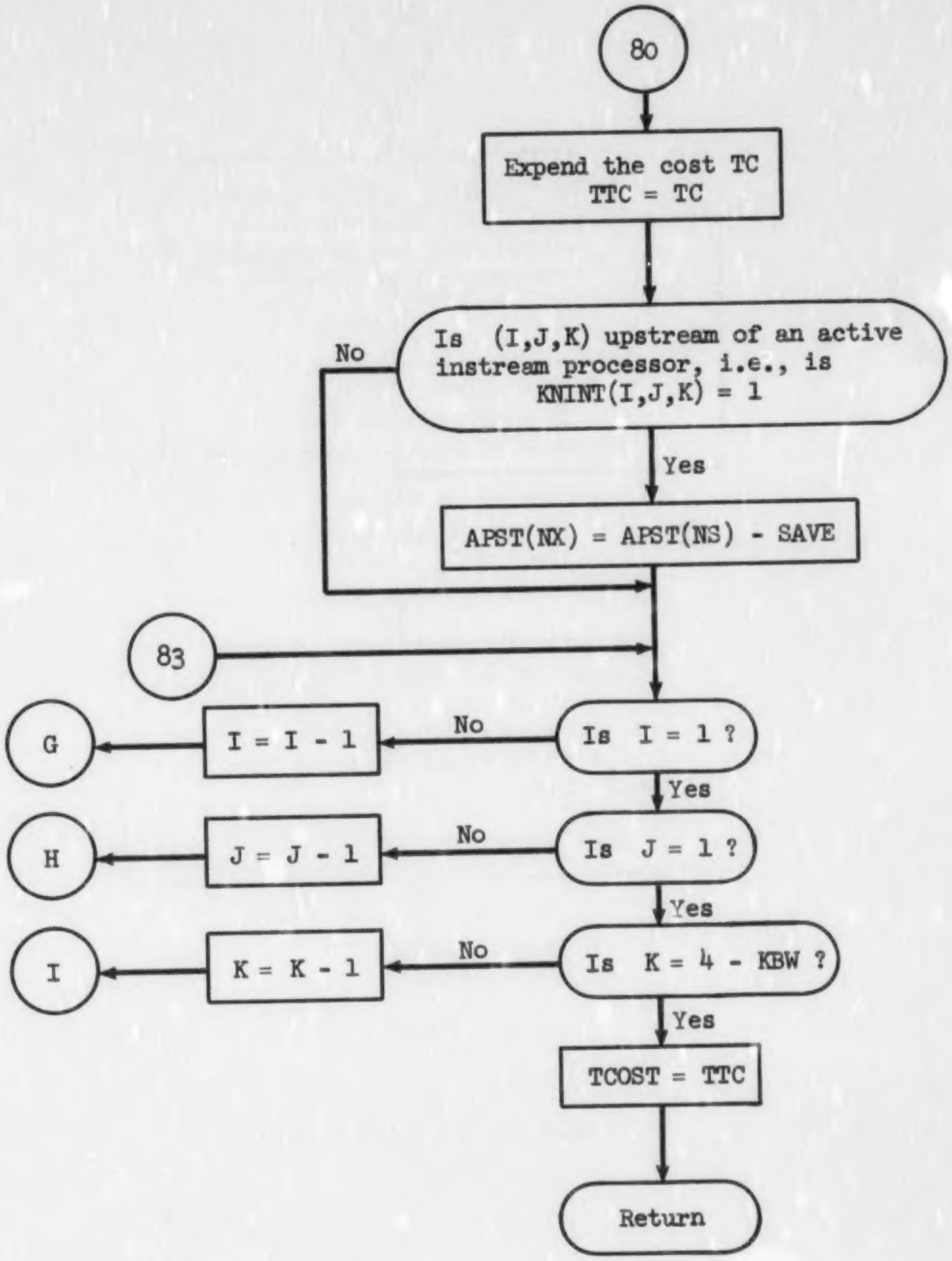

80
Expend the cost TC
TTC = TC
Is (I,J,K) upstream of an active instream processor, i.e., is KNINT(I,J,K) = 1
No
Yes
APST(NX) = APST(NS) - SAVE
83
Is I = 1 ?
No
I = I - 1
G
Yes
Is J = 1 ?
No
J = J - 1
H
Yes
Is K = 4 - KBW ?
No
K = K - 1
I
Yes
TCOST = TTC
Return

Figure C.10 Subroutine NEFESE

Subroutine TOFE(NT,II,JJ,KK)

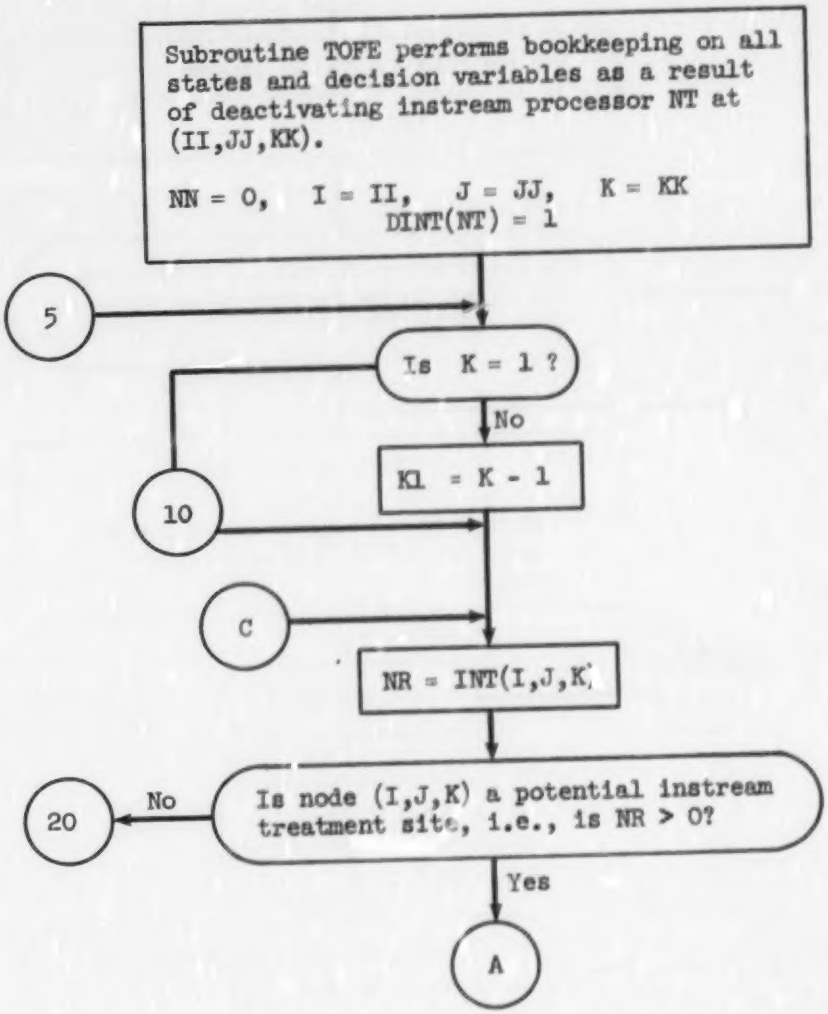

Subroutine TOFE performs bookkeeping on all states and decision variables as a result of deactivating instream processor NT at (II,JJ,KK).
NN = 0, I = II, J = JJ, K = KK
DINT(NT) = 1
5
Is K = 1 ?
No
K1 = K - 1
10
C
NR = INT(I,J,K)
20
No
Is node (I,J,K) a potential instream treatment site, i.e., is NR > 0?
Yes
A

Figure C.11 Subroutine TOFE

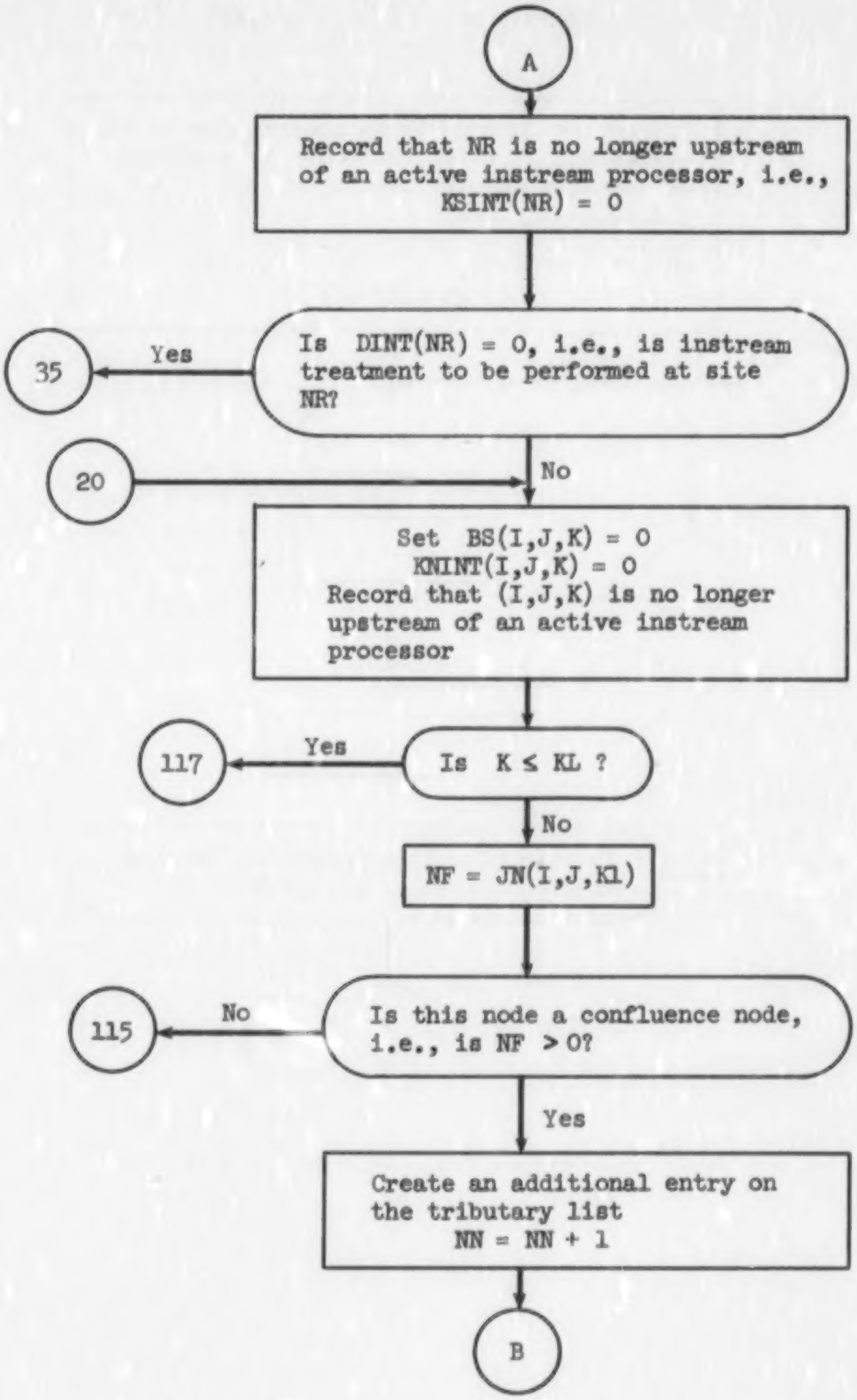

A
Record that NR is no longer upstream of an active instream processor, i.e., KSINT(NR) = 0
Is DINT(NR) = 0, i.e., is instream treatment to be performed at site NR?
Yes
35
No
20
Set BS(I,J,K) = 0
KNINT(I,J,K) = 0
Record that (I,J,K) is no longer upstream of an active instream processor
Is K ≤ KL ?
Yes
117
No
NF = JN(I,J,K1)
Is this node a confluence node, i.e., is NF > 0?
No
115
Yes
Create an additional entry on the tributary list
NN = NN + 1
B

Figure C.11 Subroutine TOFE

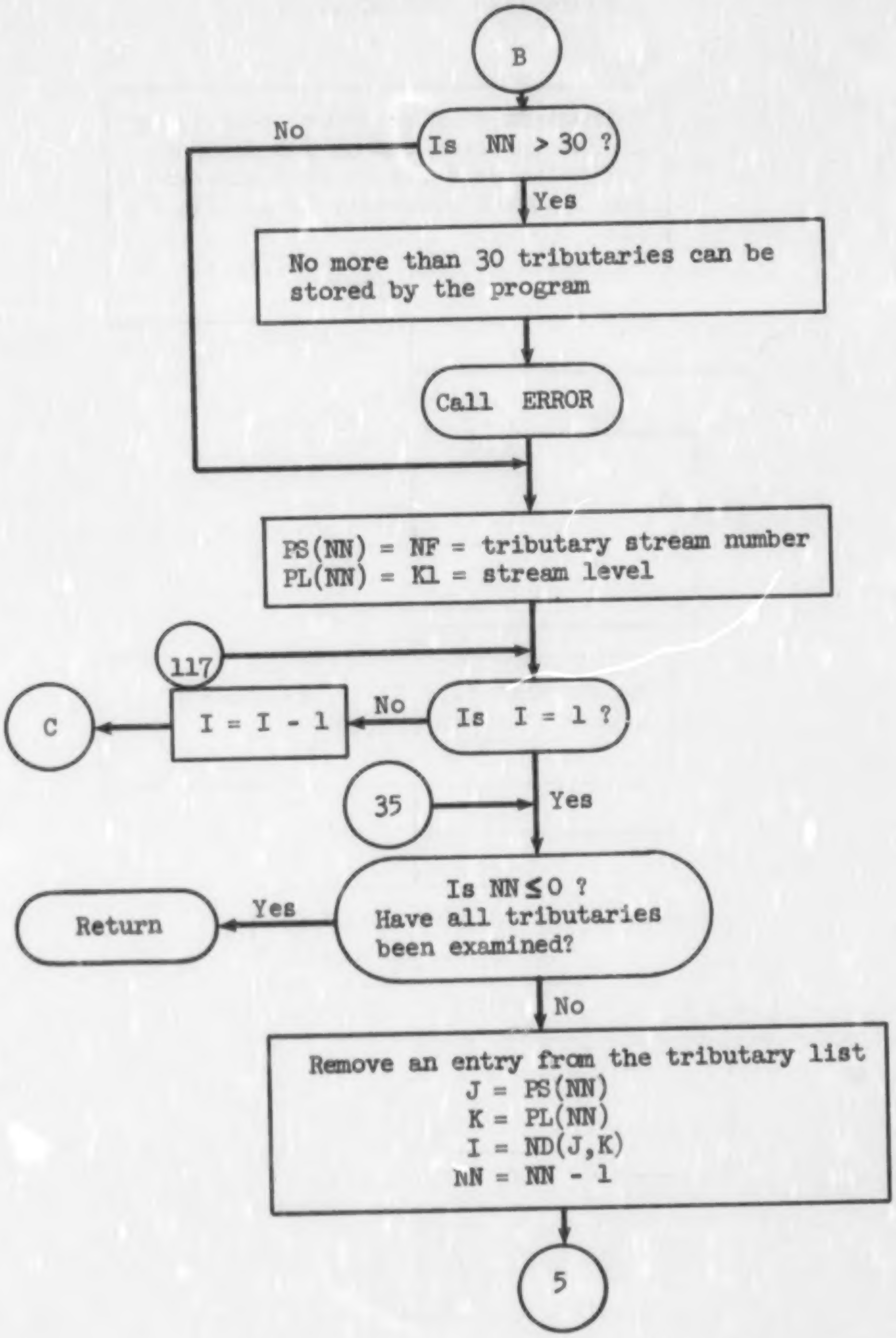

B
Is NN > 30 ?
No
Yes
No more than 30 tributaries can be stored by the program
Call ERROR
PS(NN) = NF = tributary stream number
PL(NN) = K1 = stream level
117
Is I = 1 ?
No
I = I - 1
C
35
Yes
Is NN ≤ 0 ? Have all tributaries been examined?
Yes
Return
No
Remove an entry from the tributary list
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN - 1
5

Figure C.11 Subroutine TOFE

Subroutine TONE(NT,II,JJ,KK)

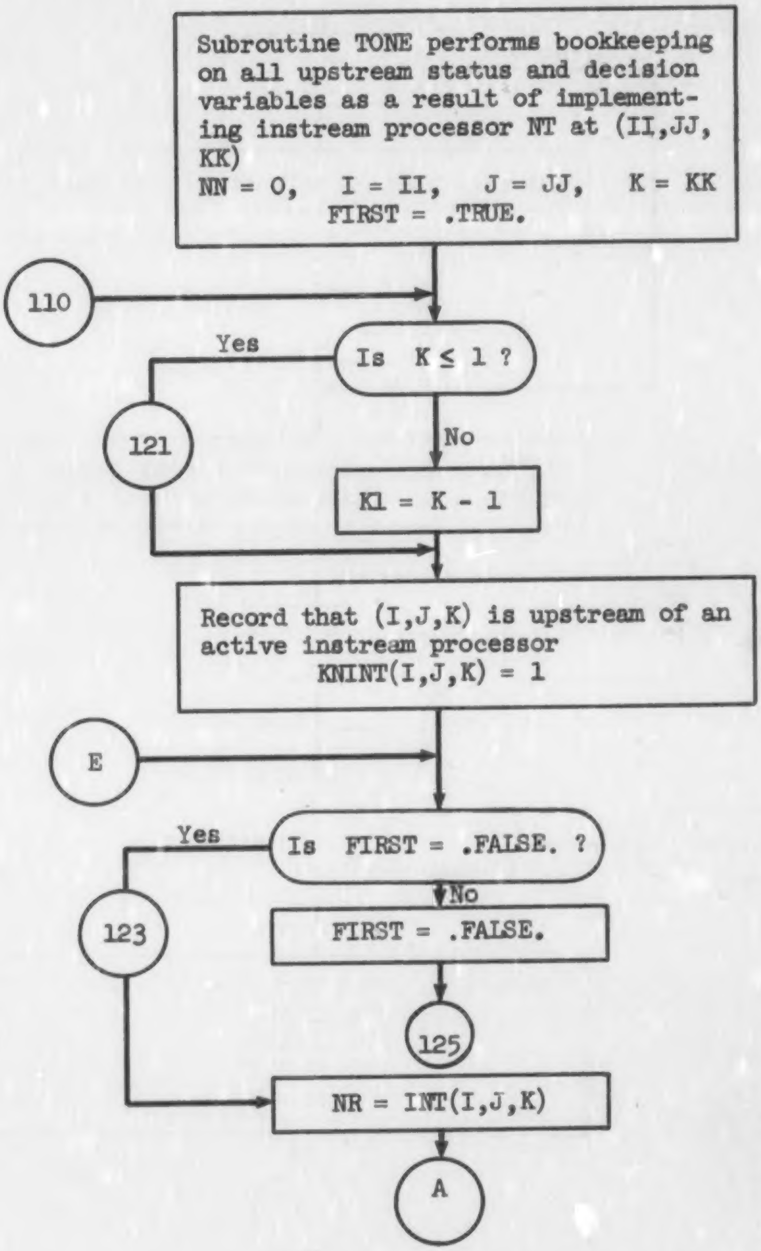

Subroutine TONE performs bookkeeping on all upstream status and decision variables as a result of implementing instream processor NT at (II,JJ, KK)
NN = 0, I = II, J = JJ, K = KK
FIRST = .TRUE.
110
Yes
Is K ≤ 1 ?
121
No
K1 = K - 1
Record that (I,J,K) is upstream of an active instream processor
KNINT(I,J,K) = 1
E
Yes
Is FIRST = .FALSE. ?
No
123
FIRST = .FALSE.
125
NR = INT(I,J,K)
A

Figure C.12 Subroutine TONE

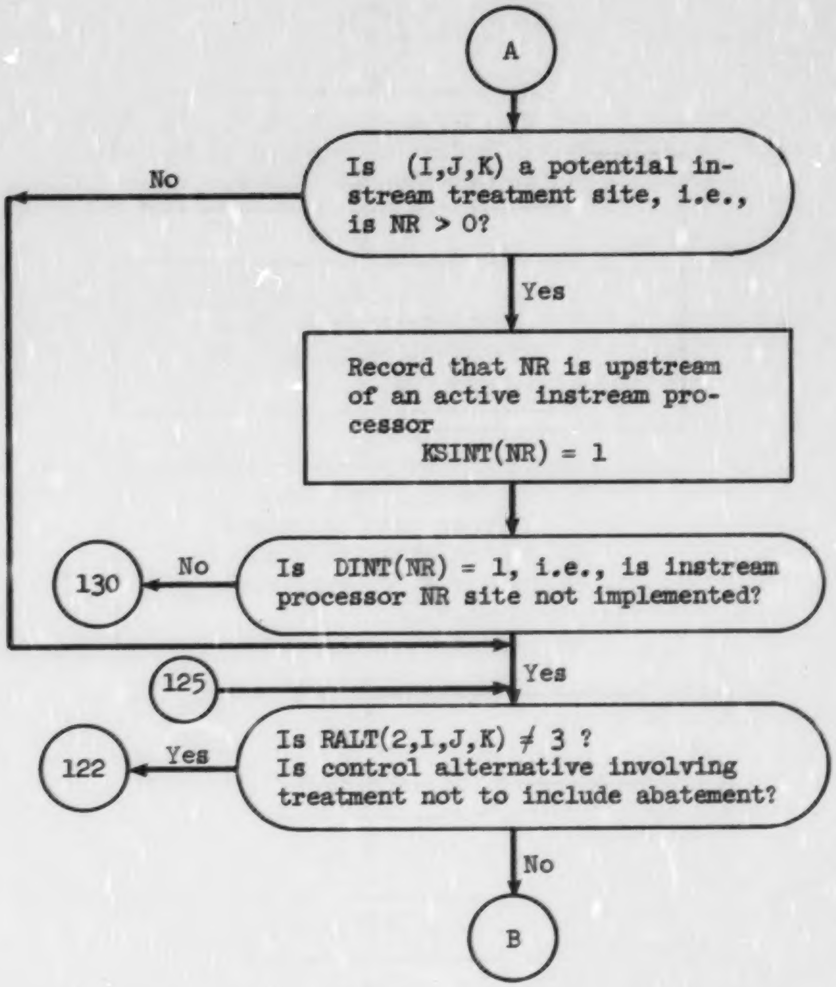

A
Is (I,J,K) a potential instream treatment site, i.e., is NR > 0?
No
Yes
Record that NR is upstream of an active instream processor
KSINT(NR) = 1
Is DINT(NR) = 1, i.e., is instream processor NR site not implemented?
No
130
Yes
125
Is RALT(2,I,J,K) ≠ 3 ?
Is control alternative involving treatment not to include abatement?
Yes
122
No
B

Figure C.12 Subroutine TONE

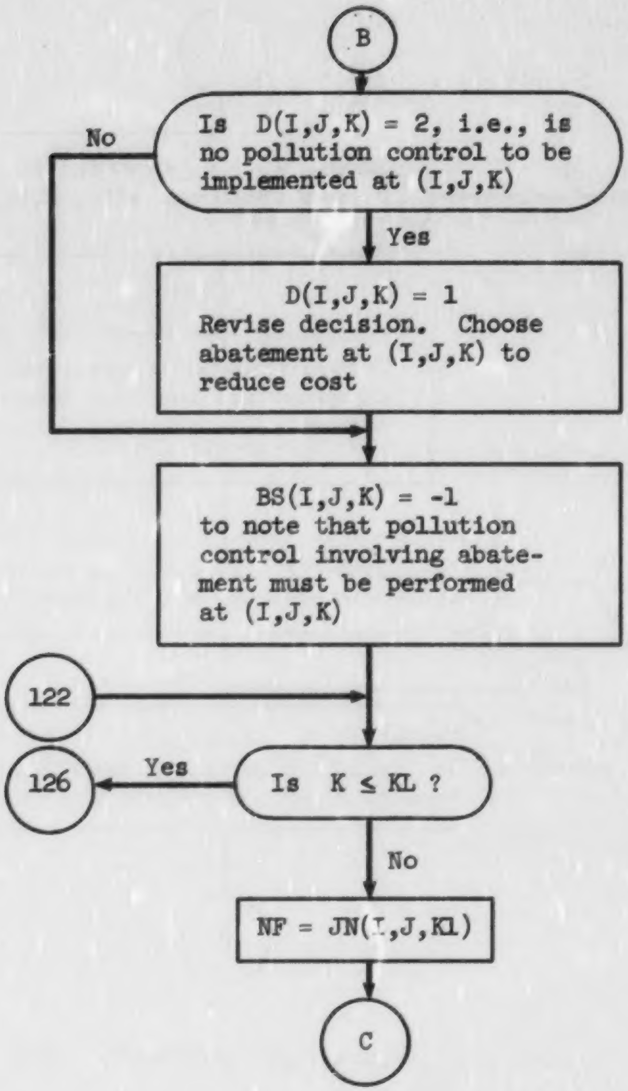

B
Is D(I,J,K) = 2, i.e., is no pollution control to be implemented at (I,J,K)
No
Yes
D(I,J,K) = 1
Revise decision. Choose abatement at (I,J,K) to reduce cost
BS(I,J,K) = -1
to note that pollution control involving abatement must be performed at (I,J,K)
122
126
Yes
Is K ≤ KL ?
No
NF = JN(I,J,K1)
C

Figure C.12 Subroutine TONE

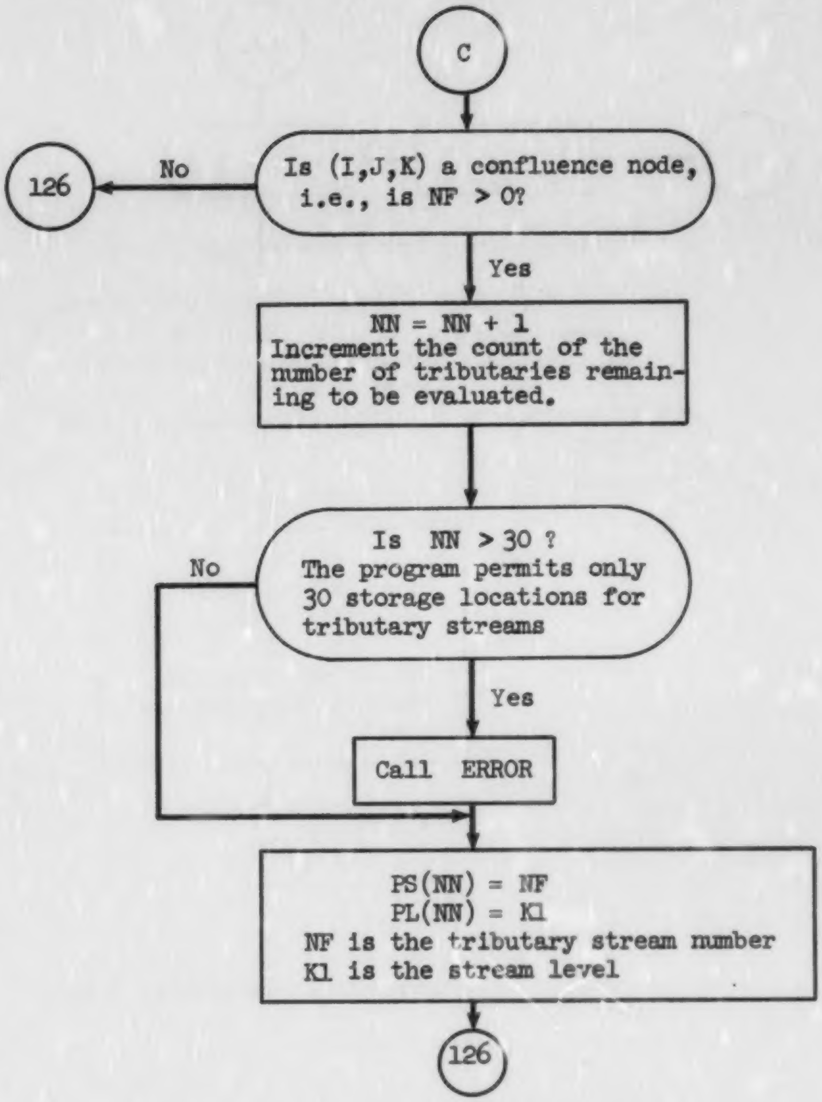

C
No
126
Is (I,J,K) a confluence node, i.e., is NF > 0?
Yes
NN = NN + 1
Increment the count of the number of tributaries remaining to be evaluated.
Is NN > 30 ?
The program permits only 30 storage locations for tributary streams
No
Yes
Call ERROR
PS(NN) = NF
PL(NN) = K1
NF is the tributary stream number
K1 is the stream level
126

Figure C.12 Subroutine TONE

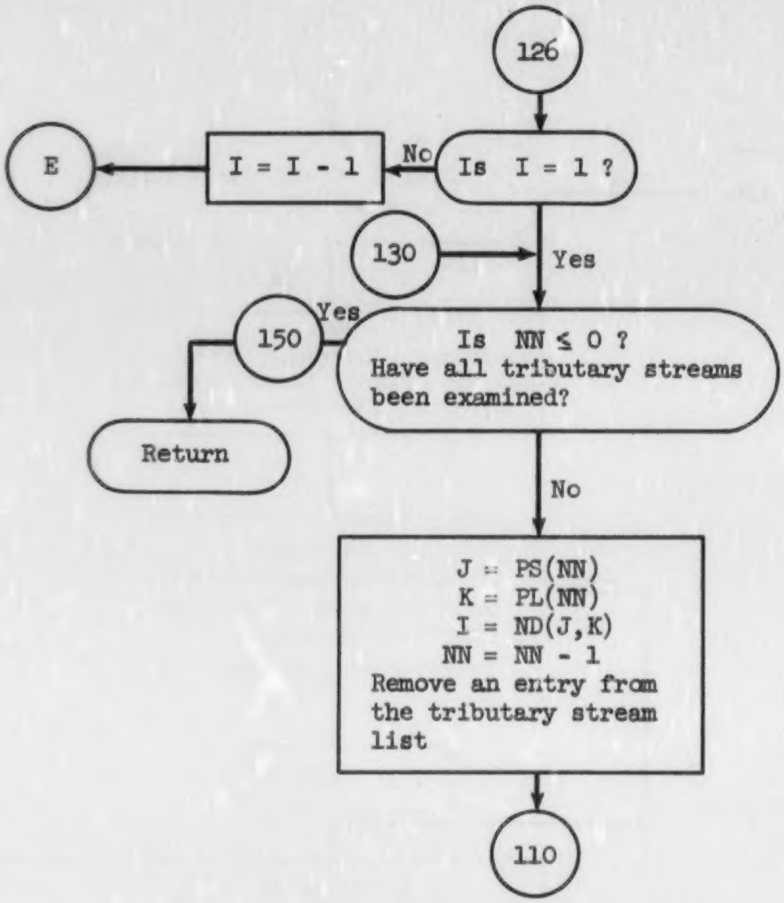

126
Is I = 1 ?
No
I = I - 1
E
Yes
130
Yes
150
Is NN ≤ 0 ?
Have all tributary streams been examined?
Return
No
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN - 1
Remove an entry from the tributary stream list
110

Figure C.12 Subroutine TONE

Subroutine ZOE(I,J,K)

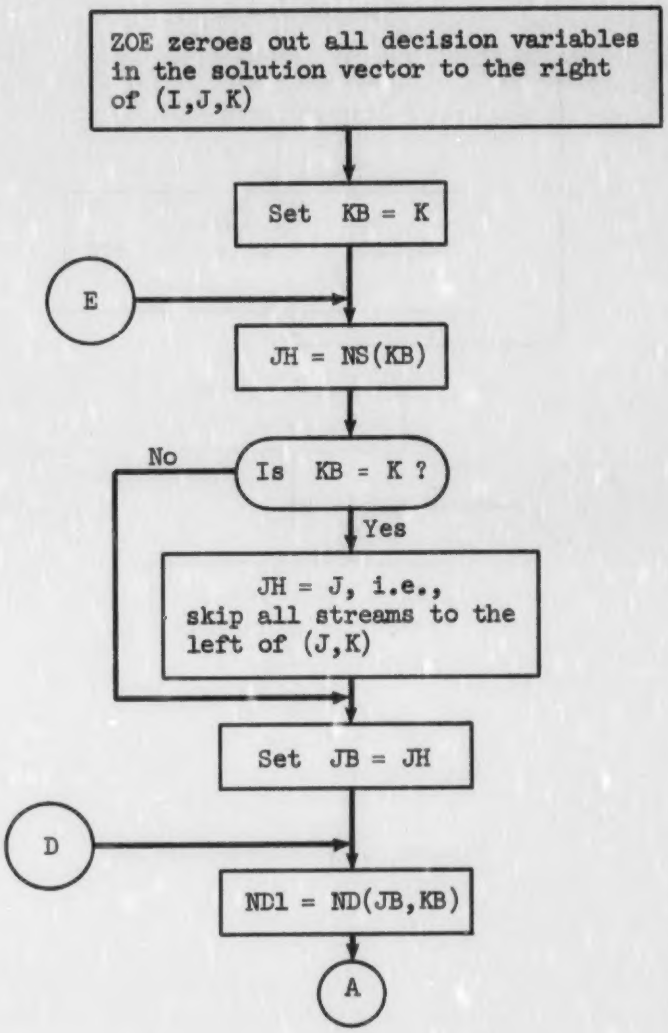

ZOE zeroes out all decision variables in the solution vector to the right of (I,J,K)
Set KB = K
E
JH = NS(KB)
No
Is KB = K ?
Yes
JH = J, i.e., skip all streams to the left of (J,K)
Set JB = JH
D
ND1 = ND(JB,KB)
A

Figure C.13 Subroutine ZOE

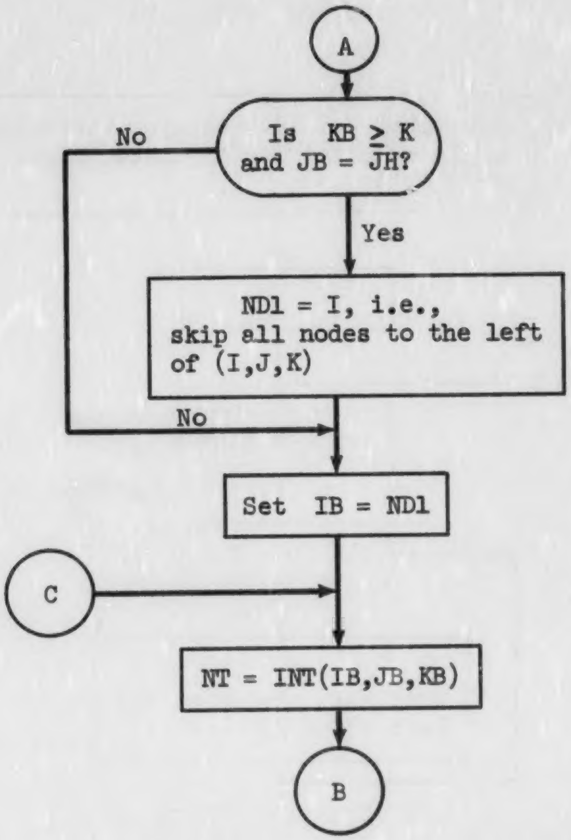
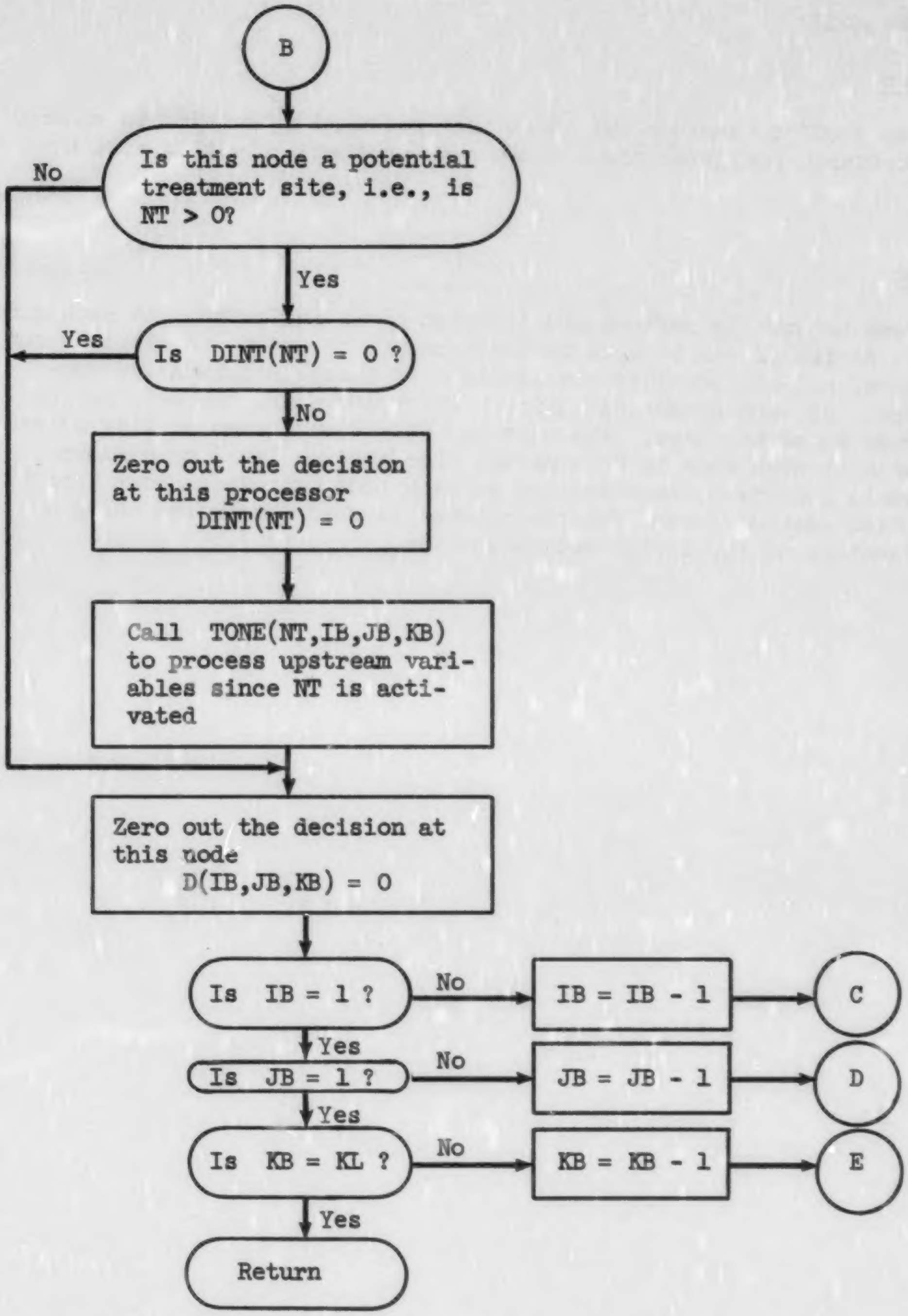

A
Is KB ≥ K
and JB = JH?
No
Yes
ND1 = I, i.e.,
skip all nodes to the left
of (I,J,K)
No
Set IB = ND1
C
NT = INT(IB,JB,KB)
B

Figure C.13 Subroutine ZOE



B
Is this node a potential treatment site, i.e., is NT > 0?
No
Yes
Is DINT(NT) = 0 ?
Yes
No
Zero out the decision at this processor DINT(NT) = 0
Call TONE(NT,IB,JB,KB) to process upstream variables since NT is activated
Zero out the decision at this node D(IB,JB,KB) = 0
Is IB = 1 ?
No
IB = IB - 1
C
Yes
Is JB = 1 ?
No
JB = JB - 1
D
Yes
Is KB = KL ?
No
KB = KB - 1
E
Yes
Return

Figure C.13 Subroutine ZOE

PROGRAM ALCOT

Purpose

Program ALCOT determines the least cost resource allocation to control mine drainage pollution for a fixed quality standard within a watershed.

Method

A stream network is defined and decision nodes indicated. At each mine source decisions can be made to treat or not to treat, to abate or not to abate, and all possible combinations of these decisions are considered. At each potential instream processor site, the site may be implemented or not used. The cost and effect on stream quality of each decision at each node is determined. For a given level of maximum allowable pollution concentration at each node, the least cost feasible pollution control scheme for the network is then determined using a modification of the Lawler-Bell algorithm.

Definition of Variables

AP(I,J,K) = Annual pollutant load emitted from source I on stream J of level K(kg).

AP1(I,J,K) = Annual pollutant load emitted from souce I on stream I of level K when source allocation alternative 1 is selected (kg).

APA(I,J,K) = Annual pollutant loading emitted from source I on stream J of level K after abatement (kg).

APN(I,J,K) = Annual pollutant load at node I on stream J of level K due to natural sources (kg).

BS(I,J,K) = 1 if source (I,J,K) is not being examined.
-1 if some form of pollution control must be performed at this source
0 otherwise.

BT(NT) = 1 if treatment site NT is not being examined.
0 if otherwise.

C(1,I,J,K) = Cost to abate source I on stream J of level K.

C(2,I,J,K) = Fixed cost to treat at source I on stream J of level K.

CI(NT) = Fixed cost to perform instream treatment at instream treatment site NT.

CALT(ID,I,J,K) = Cost of resource allocation alternative ID for source (I,J,K).

CIST(NT) = Minimum total cost for solution upstream to treatment processor NT.

D(I,J,K) = Allocation alternative selected for source (I,J,K).

D(I,J,K) = 2 if treatment is to be performed at source (I,J,K).
1 if lowest cost pollution control alternative is to be performed at source (I,J,K).
0 if no pollution control measures are to be performed at source (I,J,K).

DINT(NT) = 1 if instream treatment is to be performed at instream treatment site number NT.
0 if otherwise.

ID = 1 for lowest cost alternative.
2 for alternative involving source treatment.

INT(I,J,K) = NT if node (I,J,K) is a treatment node where NT is the treatment site number.
0 if otherwise.

IO(I,J,K,NT) = Optimal values of D array for source (I,J,K) and upstream solution to instream processor NT.

IOI(NT,I) Optimal values of DINT array for treatment site NT and upstream solution to instream treatment processor I.

JN(I,J,K) = 0 if node I on level K+1 stream J is not a confluence node.
NF otherwise where NF is the stream of level K feeding node I on level K+1 stream J.

KBW = KU-KL+1

KL = Lowest level stream represented, (KL=1 if 3 stream levels are used, KL=2 if 2 stream levels are used, KL=3 if 1 stream level is used).

KLA = KL+1.

KNT = Number of times the criterion function has been evaluated.

KOPT = Value of KNT when the optimal solution was evaluated.

KOUT = the interval between output of solution vectors

KSINT(NT) = 1 if instream processor site NT is upstream of an active instream processor.
0 if otherwise.

KU = Highest level stream (must be 3)

MDIS(I,K) = Maximum number of pollutant flows that can be stored for confluence nodes receiving flow from level K streams. I = 1 for output from the level K stream, I = 2 for upstream flow on the level K+1 stream.

MNIST = Dimensioned value of the first subscript of the IOI array.

MNO = Dimensioned value of the node number subscript of all arrays subscripted by node number.

MNS = Dimensioned value of the stream number subscript of all arrays subscripted by node number.

MS = 0 if neither abatement nor treatment is to be performed.
1 if abatement but no source treatment is to be performed.
2 if source treatment but no abatement is to be performed.
3 if both abatement and source treatment are to be performed.

ND(J,K) = Total number of nodes on stream J of level K

NDIS(I,J,K) = Number of admissible pollutant flows calculated for the confluence node receiving flow from level K stream J. I = 1 for level K stream J output flow, I = 2 for upstream flow on level K+1 stream receiving flow from level K stream J.

NFN(J,K) = Confluence node on level K+1 stream receiving flow from level K stream J.

NFS(J) = Level 2 stream receiving flow from level I stream J.

NINST = Total number of possible instream treatment site locations.

NS(K) = Total number of streams of level K.

O(I,J,K) = Optimal allocation alternative selected for source (I,J,K).

OINT(NT) = Optimal value of DINT(NT).

OI(NT) = 1 if this instream treatment has had an optimal upstream solution calculated.
0 if otherwise.

P(I,J,K) = Pollutant loading emitted from source I on stream J of level K (kg/hr).

P1(I,J,K) = Pollutant output for resource allocation alternative 1 for source (I,J,K) (kg/hr).

PL(I) = Level of Ith stream to process.

PLT(I,J,K) =	Total pollutant load just downsteam of node I on stream J of level K given resource allocation specified by D and DINT arrays (kg/hr).
PN(I,J,K) =	Cumulative natural pollutant load occurring at node (I,J,K) without mine drainage assuming all instream processors are used (kg/hr). Note that input values are incremental flows between (I,J,K) and the next upstream node.
PNT(I,J,K) =	Natural pollutant flow at node (I,J,K) assuming no instream processors are used.
PS(I) =	Ith stream to process.
PT(J,K) =	Pollutant input from stream J of level K,K=1,2
Q(I,J,K) =	Stream flow at node (I,J,K) excluding the pollutant (input as cubic meters per second converted to kg/hr).
QS =	Quality standard expressed as maximum pollutant concentration (ppm).
RALT(ID,I,J,K) =	Value of MS for source (I,J,K) for resource allocation alternative ID.
TC =	Optimal value of total cost.
TTC =	Trial total cost value
VC =	Annual variable cost to treat one unit of pollution ($/kg).

Input Data:

Card Number	Variable Name	Columns Used	Format
1	NS(1)	1-5	Integer
1	NS(2)	6-10	Integer
1	NS(3)	11-15	Integer
1	VC	16-25	Real
1	MNO	26-30	Integer
1	MNS	31-35	Integer
1	QS	36-45	Real
1	NINST	46-50	Integer
1	MNIST	51-55	Integer
1	KOUT	56-60	Integer
2	ND(1,1)	1-5	Integer
2	ND(2,1)	6-10	Integer
2	ND(3,1)	11-15	Integer
	⋮		
2	ND(NS(1),1)	(5NS(1)-4)-5NS(1)	Integer
4-KL	ND(1,2)	1-5	Integer
	⋮		
4-KL	ND(NS(2),2)	(5NS(2)-4)-5NS(2)	Integer
5-KL	ND(1,3)	1-5	Integer
(see note 1)	⋮	⋮	⋮
5-KL	ND(NS(3),3)	(5NS(3)-4)-5NS(3)	Integer
6-KL	JN(1,1,1)	1-5	Integer
	JN(2,1,1)	6-10	Integer
	JN(3,1,1)	11-15	Integer
	⋮		
6-KL	JN(ND(1,2),1,1)	(5ND(1,2)-4)-5ND(1,2)	Integer
	JN(1,2,1)	1-5	Integer
	JN(2,2,1)	6-10	Integer
(see note 2)	⋮		
6+NS(2)-KL	JN(ND(1,3),1,2)	(5ND(1,3)-4)-5ND(1,3)	Integer
7+NS(2)-KL	P(1,1,1)	1-10	Real
	PA(1,1,1)	11-20	Real
	AP(1,1.1)	21-30	Real
	APA(1,1,1)	31-40	Real
	Q(1,1,1)	41-50	Real
	PN(1,1,1)	51-60	Real
	APN(1,1,1)	61-70	Real
7+NS(2)-KL	INT(1,1,1)	71-80	Real
8+NS(2)-KL	C(1,1,1,1)	1-10	Real
8+NS(2)-KL	C(2,1,1,1)	11-20	Real

9+NS(2)-KL	P(2,1,1)	1-10	Real
	PA(2.1.1)	11-20	Real
	AP(2,1,1)	21-30	Real
	APA(2,1,1)	31-40	Real
	Q(2,1,1)	41-50	Real
	PN(2,1,1)	51-60	Real
	APN(2,1,1)	61-70	Real
	INT(2,1,1)	71-80	Real
10+NS(2)-KL	C(1,2,1,1)	1-10	Real
10+NS(2)-KL (see note 3)	C(2,2,1,1)	11-20	Real
6+2w+NS(2)-KL	CI(1)	1-10	Real
	CI(2)	11-20	Real
	⋮		
6+2w+NS(2)-KL+NIN (see note 4)	CI(NINST)	(10NINST-9-80NIN-10NINST-80NIN)	Real
7+2w+NS(2)-KL+NIN (see note 5)	KOPT	1-10	Integer
	KNT	11-20	Integer
	KNTLIM	21-30	Integer
8+2w+NS(2)-KL+NIN (see note 6)	DL,DINTL,TC, D,BS,O,IO,IOI, DINT,BT,OI, KSINT,OINT,CIST arrays		

Notes:

1. The number of ND array cards is variable, depending on the number of stream levels. For a 3-level network there are three ND array cards; for a 2-level network there are two ND array cards; for a 1-level network there is one ND array card. These input instructions assume a 3-level network.

2. The number of JN array cards depends upon the total number of streams of all levels. JN array cards are sequenced numerically according to the lowest level streams, next lowest level, etc. Note that if there is only a level-three stream in the network, this sequence of cards is skipped. If there are only level-two streams and a level-three stream, there is only one JN array card. If there are level-one streams, a JN array card is provided for each level-two and level-three stream. The above input data instructions assume three stream levels exist.

3. There are two cards entered for each node using the formal specified for the following inputs: P(I,J,K), PA(I,J,K), AP(I,J,K), A APA(I,J,K), Q(I,J,K), PN(I,J,K), APN(I,J,K), INT(I,J,K), C(1,I,J,K), and C(2,I,J,K). The nodes are entered in sequence starting with the first node on the first stream of level KL. After the nodes for this stream are entered, the next stream of level KL is entered

one node at a time. After recording the data for all level KL streams, then level KL+1 streams are entered. The last node entered is the last node on the level 3 stream. A total of

$$w = \sum_{K=KL}^{3} \sum_{J=1}^{NS(K)} ND(J,K) \text{ nodes are entered on } 2w \text{ cards.}$$

4. $$NIN = \begin{cases} 1 \text{ if } NINST > 8 \\ 0 \text{ if otherwise} \end{cases}$$

5. For the initial run, KOPT and KNT should be set to zero.

6. This sequence of cards is not used on an initial run and is only used when a run restarts after KNT solution vectors have been evaluated. If an optimal solution is not reached within KNTLIM iterations, a sequence of cards will be punched. The first card punched contains current values of KOPT and KNT, to which a value of KNTLIM must be added for subsequent runs. The remaining cards are placed behind the one containing the new values of KOPT, KNT, KNTLIM to restart the run. The previous KOPT, KNT, KNTLIM card is removed and the program may be restarted using the new data. The new data give values to the TC, D, BS, O, IO, IOI, DINT, BT, OI, KSINT, OINT, CIST arrays as shown.

Common Areas Referenced

COMMON/ZER/MNS, BT

COMMON/NEX/MNOS, MNO, NS, ND, D, INT, DINT, BS, PLT, NFN, NFS, PN, P1, Q, P, KL, KU, KBW, KLA

COMMON/TCO/JN, PT, AP, AP1, CALT, CT, VC, APN

COMMON/PTT/PNT, MAXMU, KFN

COMMON/TOF/OI, CIST, KSINT

COMMON/TONN/NSO, RALT, IO, IOI

COMMON/DIS/NDIS, PO1DIS, PO2DIS, PJ1DIS, PJ2DIS, MDIS

Subroutines Required

Subroutine CONO - determines whether sufficient input pollution values to a confluence node have been determined to specify the maximum output from the node that is less than or equal to a specified upper limit. If so, the maximum node output less than the specified upper limit is calculated.

Subroutine ERROR - used to abort a run when undesirable conditions occur, and ERROR generates a traceback calling sequence.

Subroutine NEXFES - determines the next feasible solution.

Function NON - returns a one if node (I,J,K) is upstream of node (IM, JM,KM), and a zero otherwise.

Subroutine PTMAX - attempts to resolve an uncertain maximum at the node (IS,JS,KS) by determining the maximum flow less than or equal to PU. If another uncertain maximum is encountered, processing is stopped. PTMAX only examines the flow along the stream (JS,KS).

Subroutine PTMX - computes the maximum pollution flow rate past a given node which is less than or equal to PLTMAX. PTMX resolves uncertain maxima.

Subroutine STORE - stores confluence node pollution distribution values in array PDIS.

Function TCOST - computes the total resource cost.

Subroutine TOFF - processes upstream decision and status variables when an instream treatment processor is deleted.

Subroutine TON - processes upstream decision and status variables when an instream treatment processor is implemented.

Subroutine ZO - zeroes out all decision variables that are lower order than node (I,J,K).

Program ALCOT

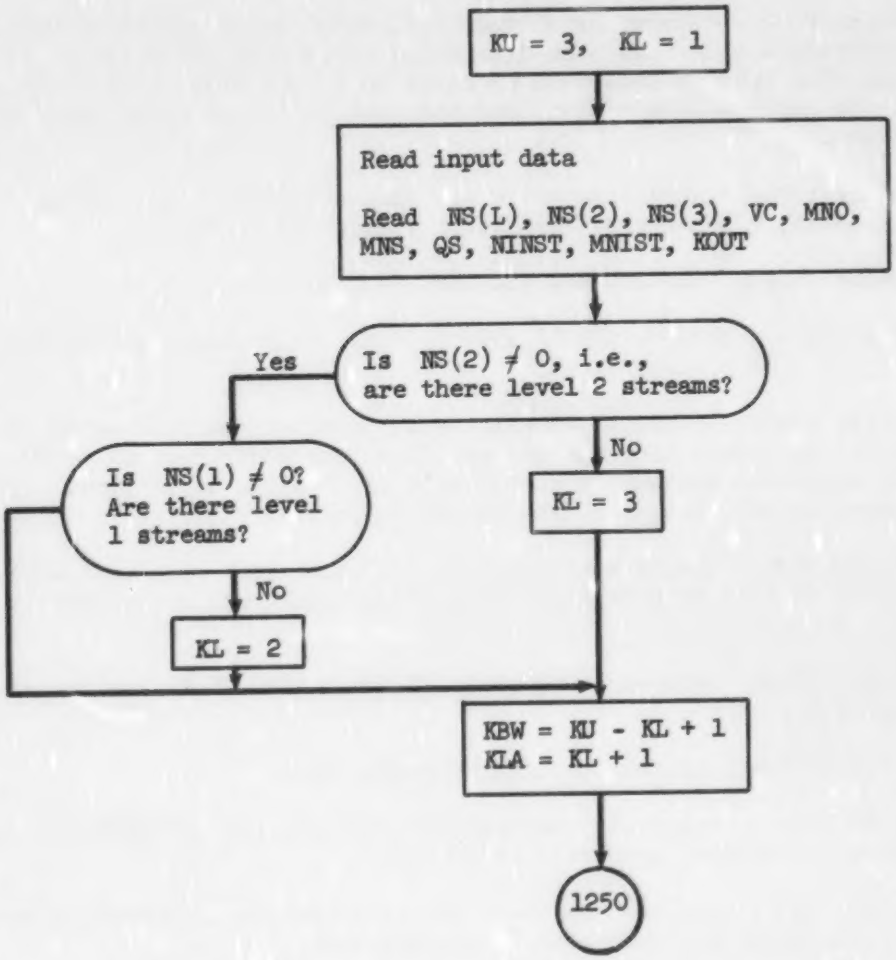

KU = 3, KL = 1
Read input data
Read NS(L), NS(2), NS(3), VC, MNO, MNS, QS, NINST, MNIST, KOUT
Is NS(2) ≠ 0, i.e., are there level 2 streams?
Yes
No
Is NS(1) ≠ 0? Are there level 1 streams?
No
KL = 3
KL = 2
KBW = KU - KL + 1
KLA = KL + 1
1250

Figure C.14 Program ALCOT

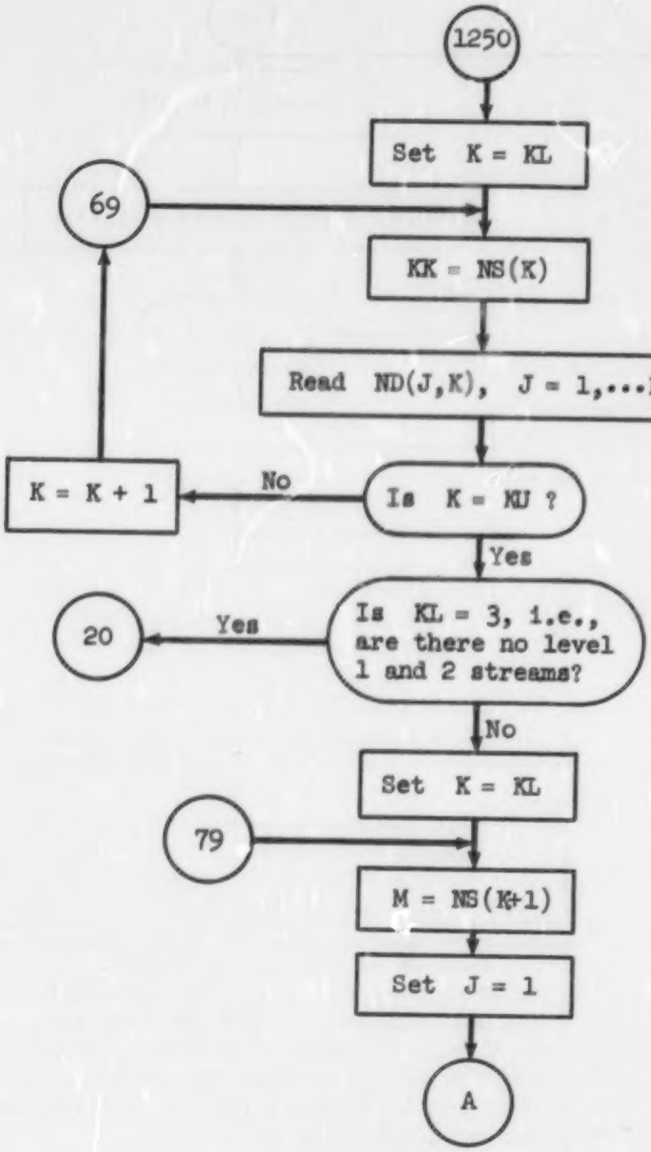
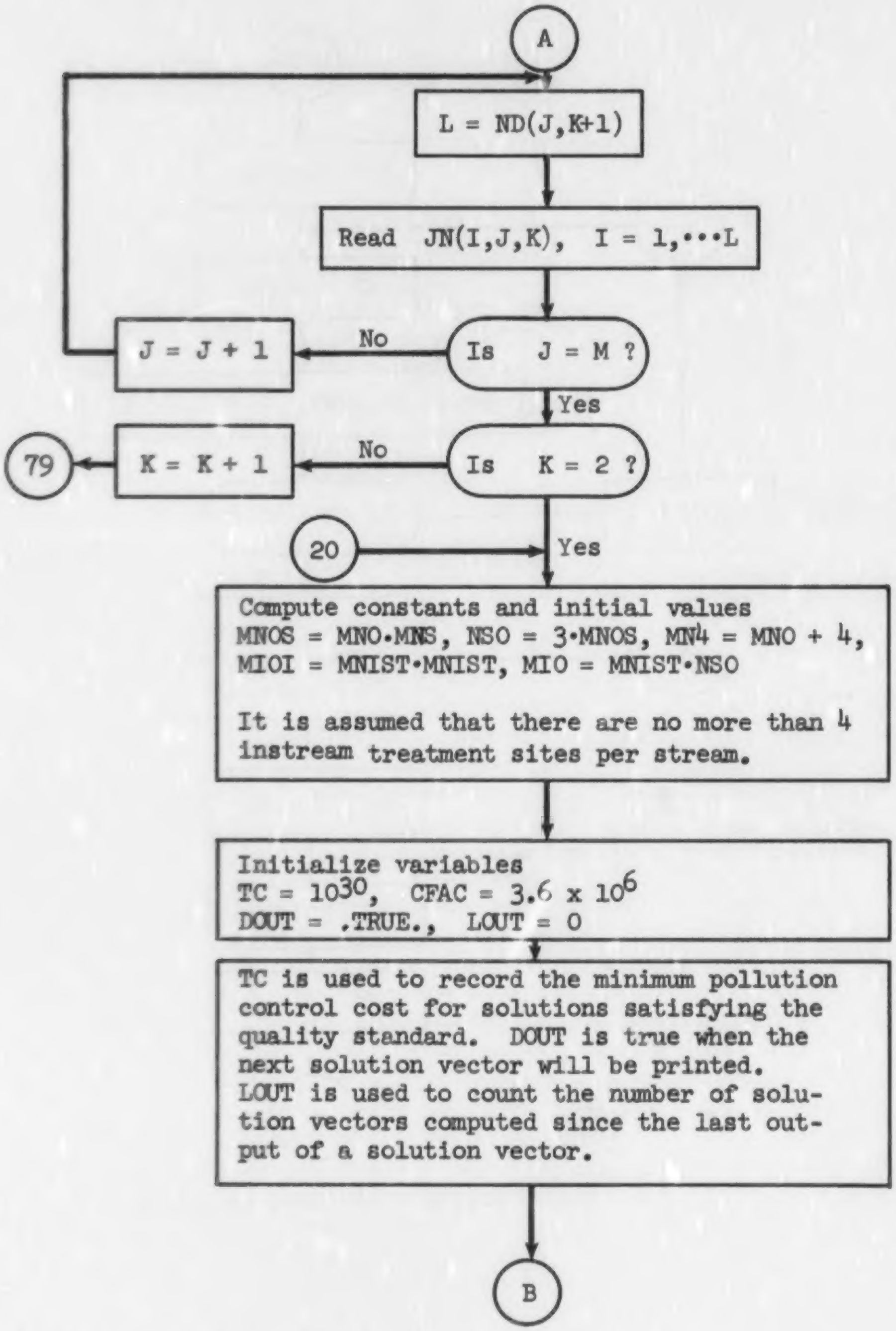

1250
Set K = KL
69
KK = NS(K)
Read ND(J,K), J = 1,...KK
Is K = KU ?
No
K = K + 1
Yes
Is KL = 3, i.e., are there no level 1 and 2 streams?
Yes
20
No
Set K = KL
79
M = NS(K+1)
Set J = 1
A

Figure C.14 Program ALCOT



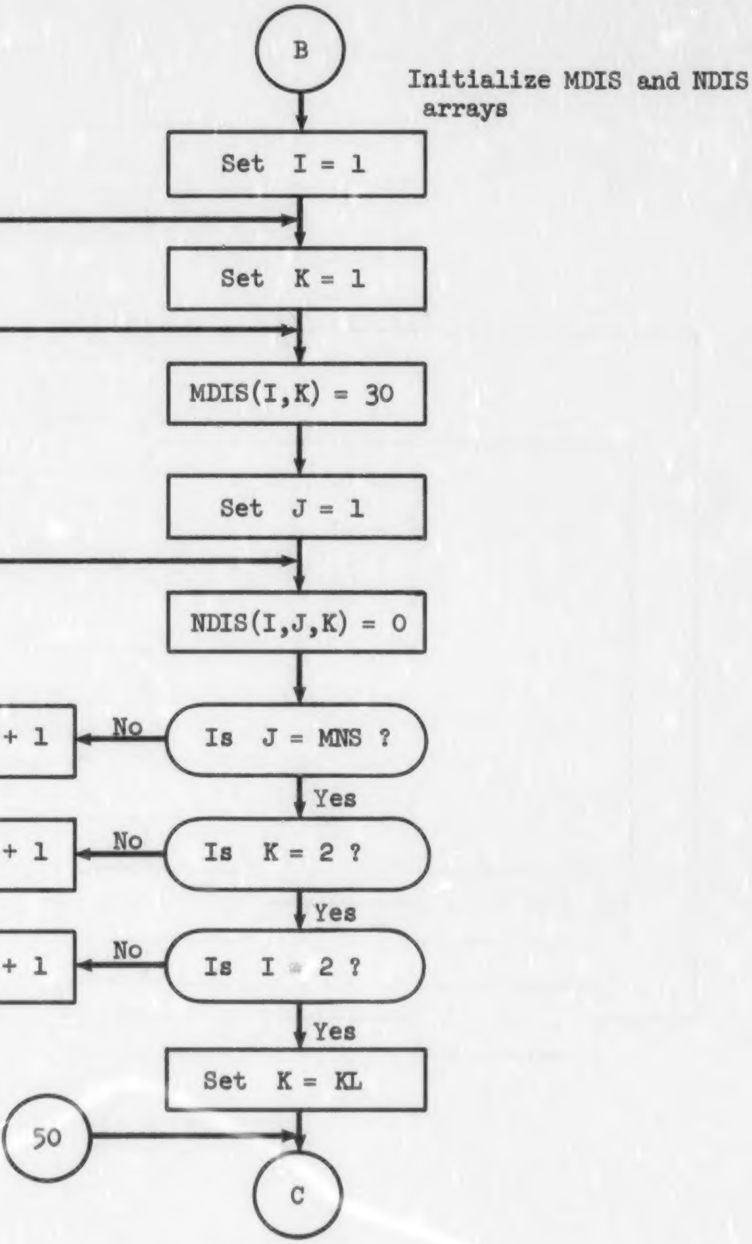
Compute constants and initial values
MNOS = MNO•MNS, NSO = 3•MNOS, MN4 = MNO + 4,
MIOI = MNIST•MNIST, MIO = MNIST•NSO

It is assumed that there are no more than 4 instream treatment sites per stream.

Initialize variables
TC = 10^{30}, CFAC = 3.6 x 10^6
DOUT = .TRUE., LOUT = 0

TC is used to record the minimum pollution control cost for solutions satisfying the quality standard. DOUT is true when the next solution vector will be printed. LOUT is used to count the number of solution vectors computed since the last output of a solution vector.

Figure C.14 Program ALCOT



B
Initialize MDIS and NDIS arrays
Set I = 1
Set K = 1
MDIS(I,K) = 30
Set J = 1
NDIS(I,J,K) = 0
Is J = MNS ?
No
J = J + 1
Yes
Is K = 2 ?
No
K = K + 1
Yes
Is I = 2 ?
No
I = I + 1
Yes
Set K = KL
50
C

Figure C.14 Program ALCOT

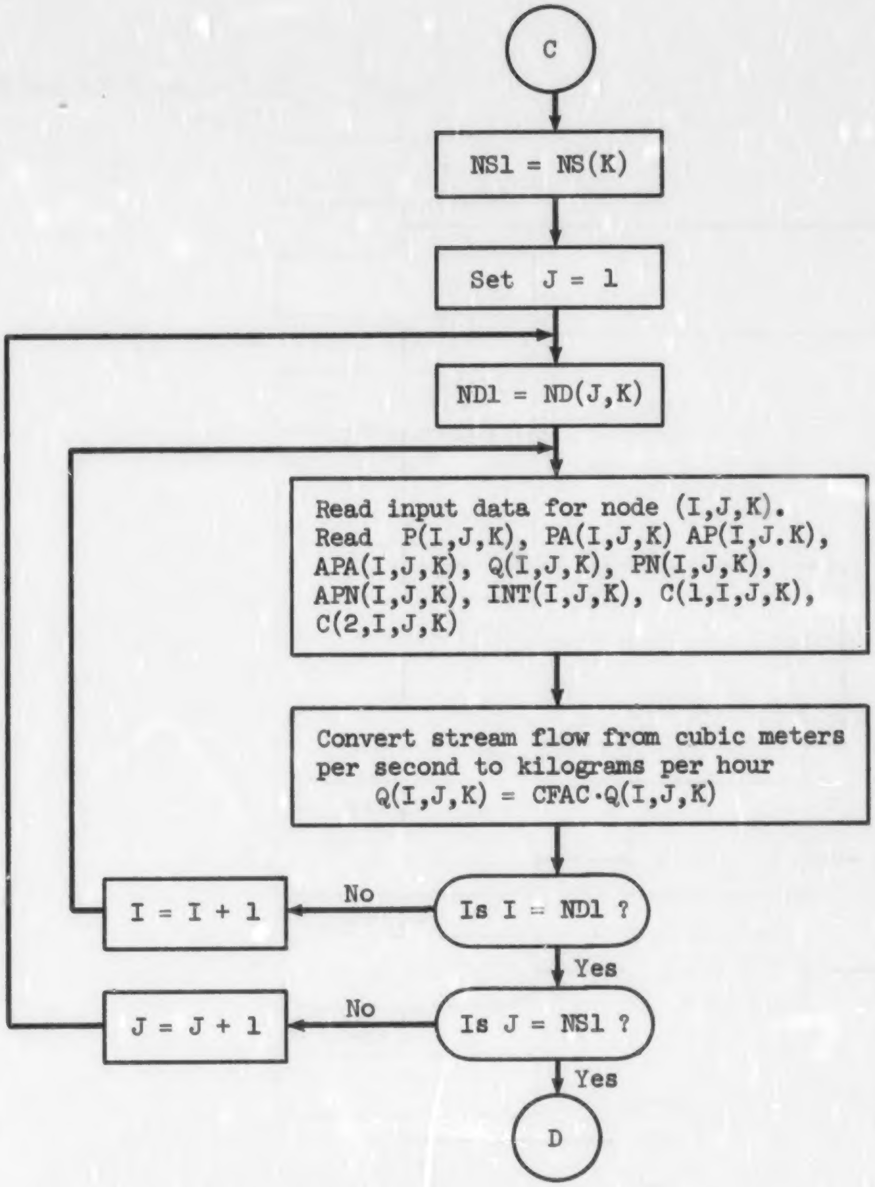

C
NS1 = NS(K)
Set J = 1
ND1 = ND(J,K)
Read input data for node (I,J,K).
Read P(I,J,K), PA(I,J,K) AP(I,J.K),
APA(I,J,K), Q(I,J,K), PN(I,J,K),
APN(I,J,K), INT(I,J,K), C(1,I,J,K),
C(2,I,J,K)
Convert stream flow from cubic meters
per second to kilograms per hour
Q(I,J,K) = CFAC·Q(I,J,K)
No
I = I + 1
Is I = ND1 ?
Yes
No
J = J + 1
Is J = NS1 ?
Yes
D

Figure C.14 Program ALCOT

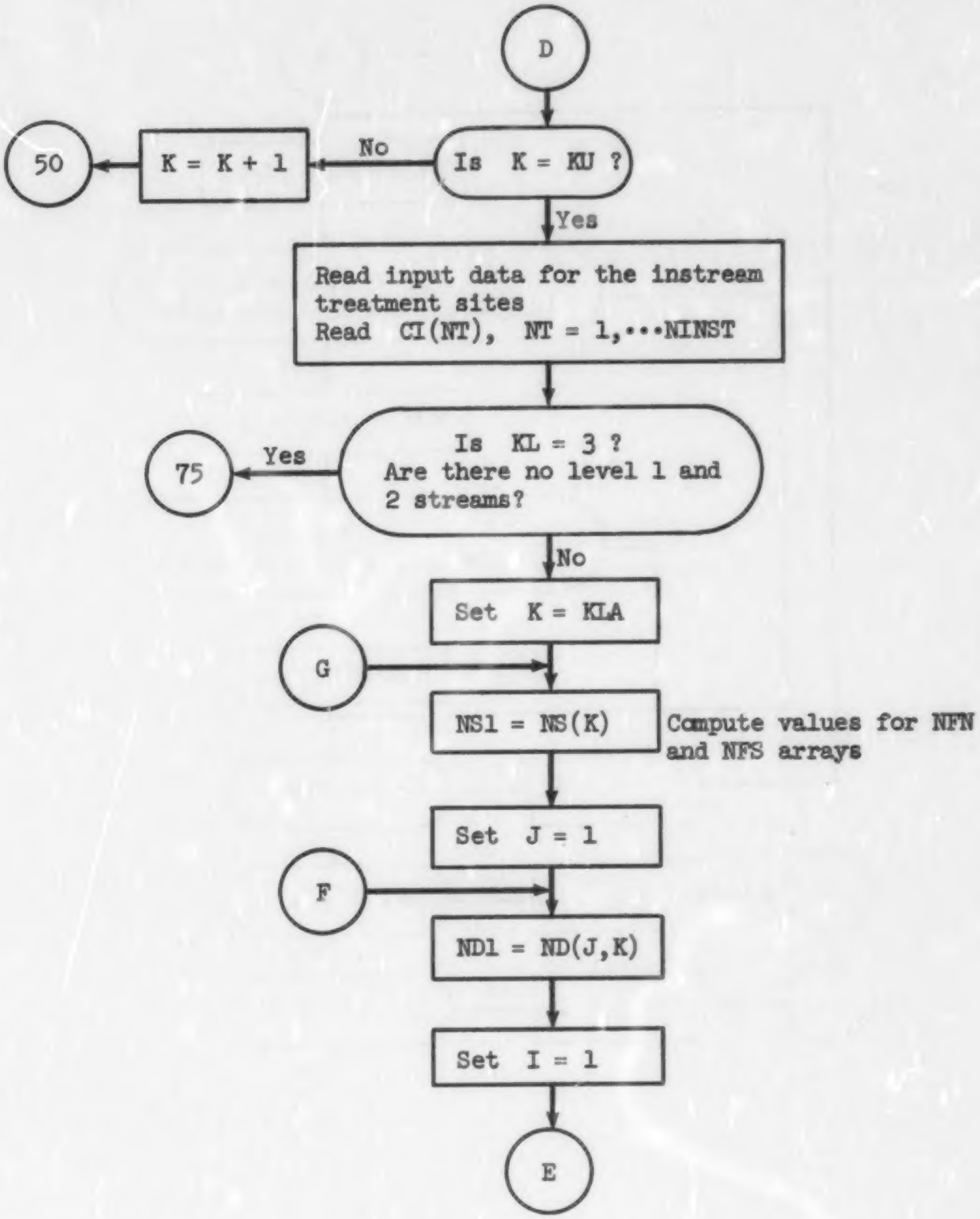

D
Is K = KU ?
No
K = K + 1
50
Yes
Read input data for the instream treatment sites
Read CI(NT), NT = 1,•••NINST
Is KL = 3 ?
Are there no level 1 and 2 streams?
Yes
75
No
Set K = KLA
G
NS1 = NS(K)
Compute values for NFN and NFS arrays
Set J = 1
F
ND1 = ND(J,K)
Set I = 1
E

Figure C.14 Program ALCOT

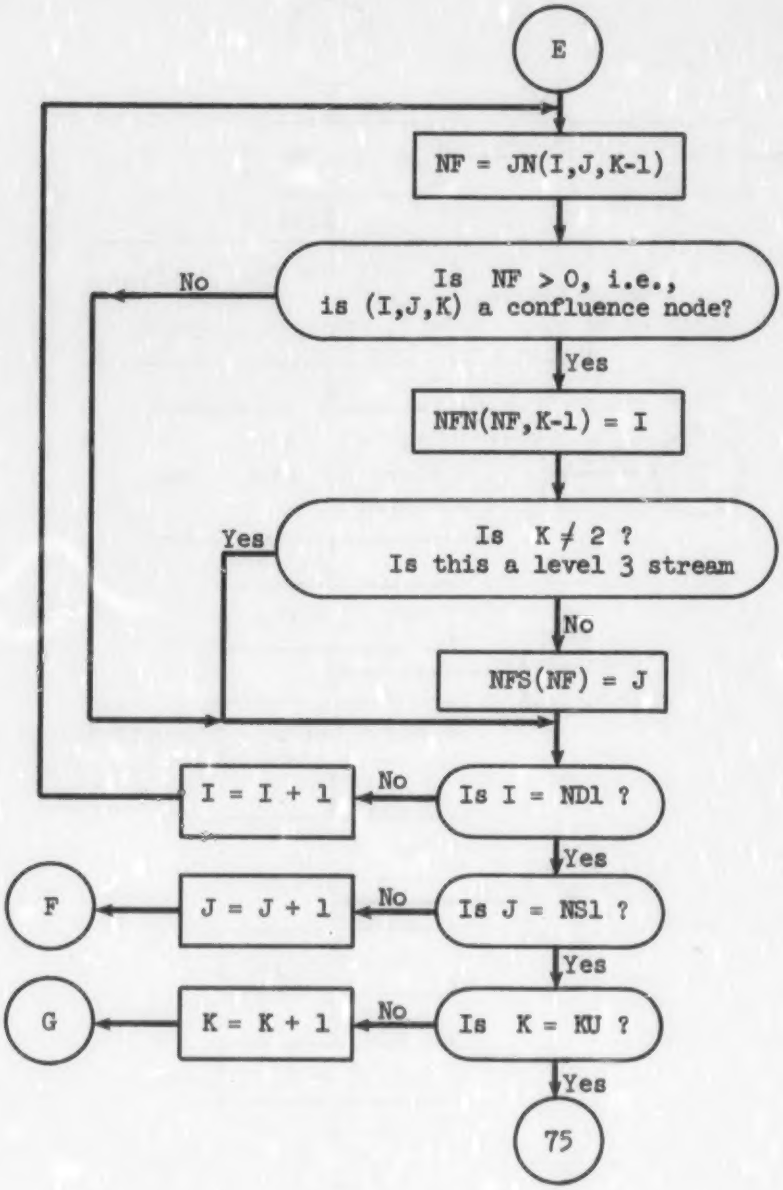

E
NF = JN(I,J,K-1)
Is NF > 0, i.e., is (I,J,K) a confluence node?
No
Yes
NFN(NF,K-1) = I
Is K ≠ 2 ? Is this a level 3 stream
Yes
No
NFS(NF) = J
I = I + 1
No
Is I = ND1 ?
Yes
F
J = J + 1
No
Is J = NS1 ?
Yes
G
K = K + 1
No
Is K = KU ?
Yes
75

Figure C.14 Program ALCOT

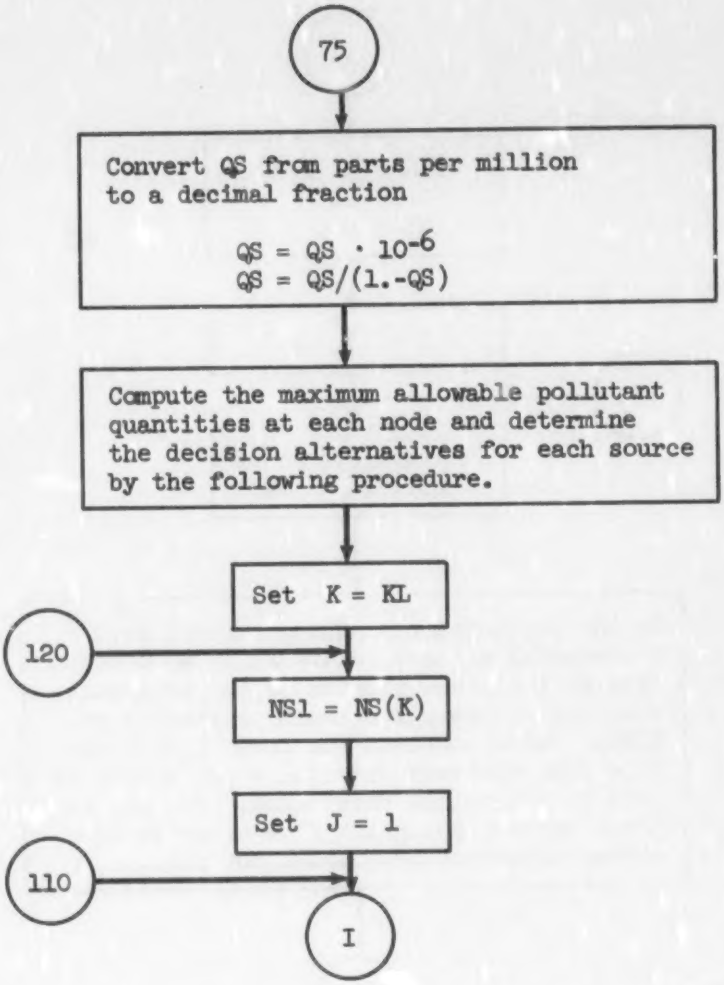

75
Convert QS from parts per million to a decimal fraction
QS = QS · 10-6
QS = QS/(1.-QS)
Compute the maximum allowable pollutant quantities at each node and determine the decision alternatives for each source by the following procedure.
Set K = KL
120
NS1 = NS(K)
Set J = 1
110
I

Figure C.14 Program ALCOT

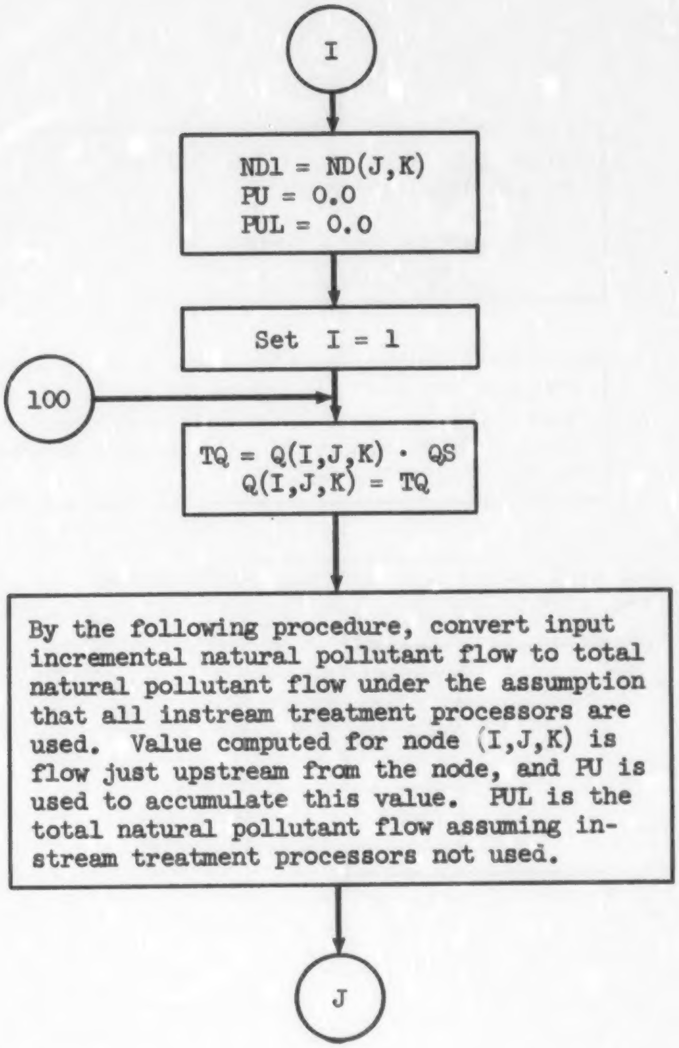

I
ND1 = ND(J,K)
PU = 0.0
PUL = 0.0
Set I = 1
100
TQ = Q(I,J,K) · QS
Q(I,J,K) = TQ
By the following procedure, convert input incremental natural pollutant flow to total natural pollutant flow under the assumption that all instream treatment processors are used. Value computed for node (I,J,K) is flow just upstream from the node, and PU is used to accumulate this value. PUL is the total natural pollutant flow assuming instream treatment processors not used.
J

Figure C.14 Program ALCOT

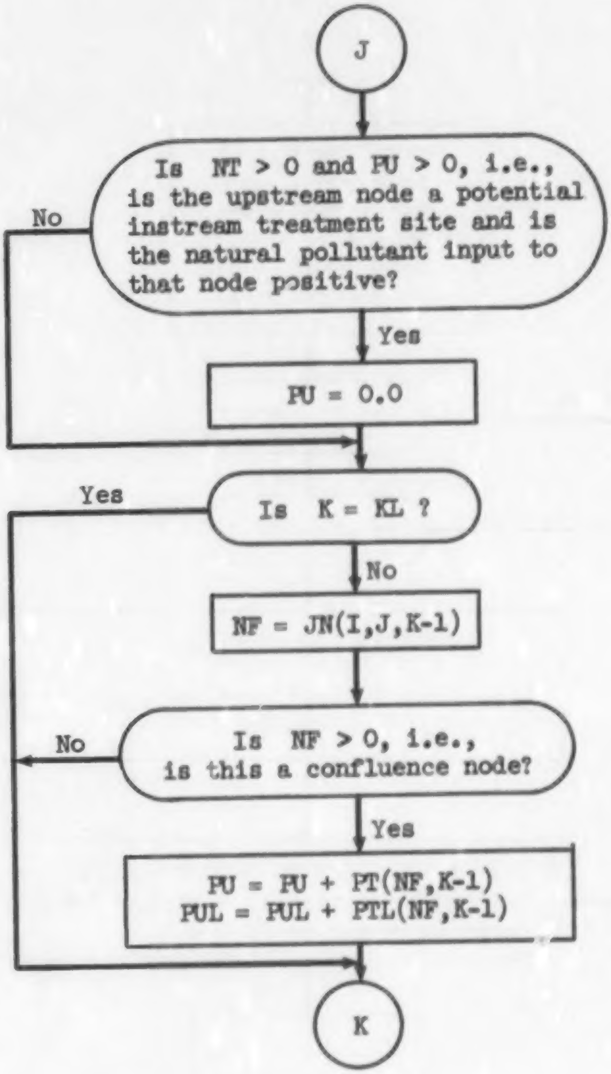

J
Is NT > 0 and PU > 0, i.e., is the upstream node a potential instream treatment site and is the natural pollutant input to that node positive?
No
Yes
PU = 0.0
Is K = KL ?
Yes
No
NF = JN(I,J,K-1)
Is NF > 0, i.e., is this a confluence node?
No
Yes
PU = PU + PT(NF,K-1)
PUL = PUL + PTL(NF,K-1)
K

Figure C.14 Program ALCOT

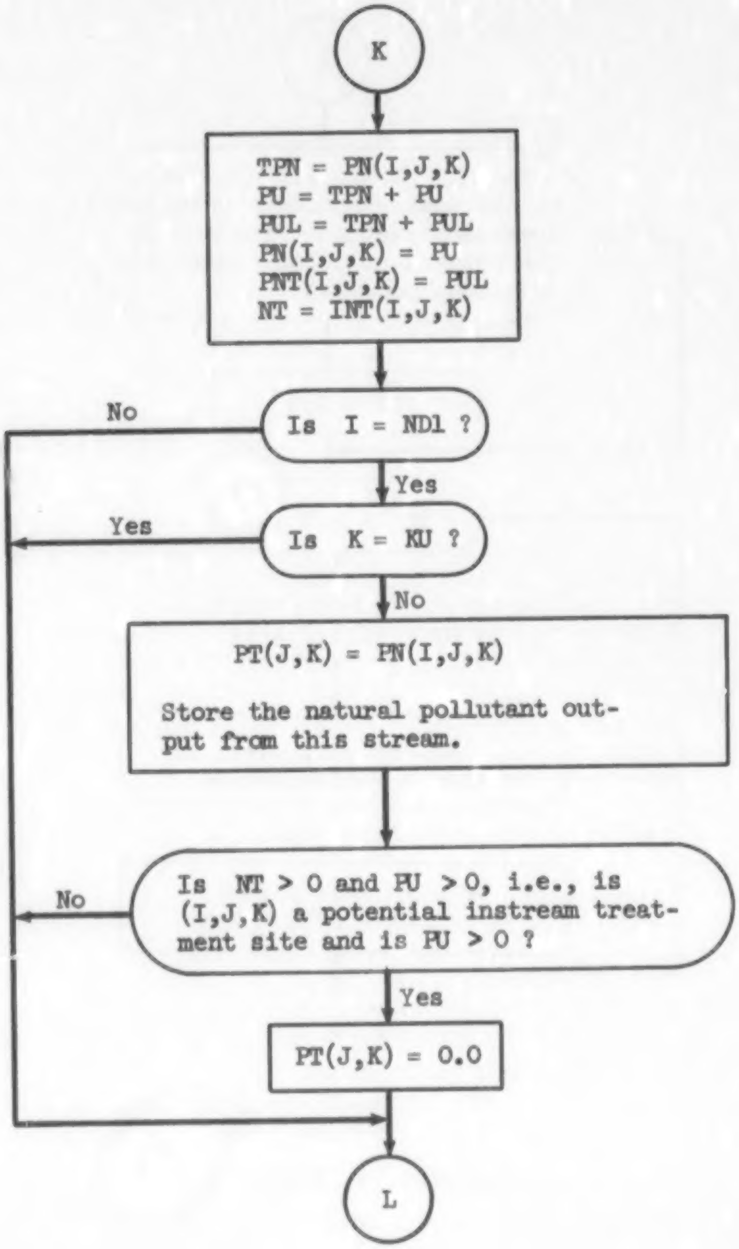

K
TPN = PN(I,J,K)
PU = TPN + PU
PUL = TPN + PUL
PN(I,J,K) = PU
PNT(I,J,K) = PUL
NT = INT(I,J,K)
Is I = ND1 ?
No
Yes
Is K = KU ?
Yes
No
PT(J,K) = PN(I,J,K)
Store the natural pollutant output from this stream.
Is NT > 0 and PU > 0, i.e., is (I,J,K) a potential instream treatment site and is PU > 0 ?
No
Yes
PT(J,K) = 0.0
L

Figure C.14 Program ALCOT

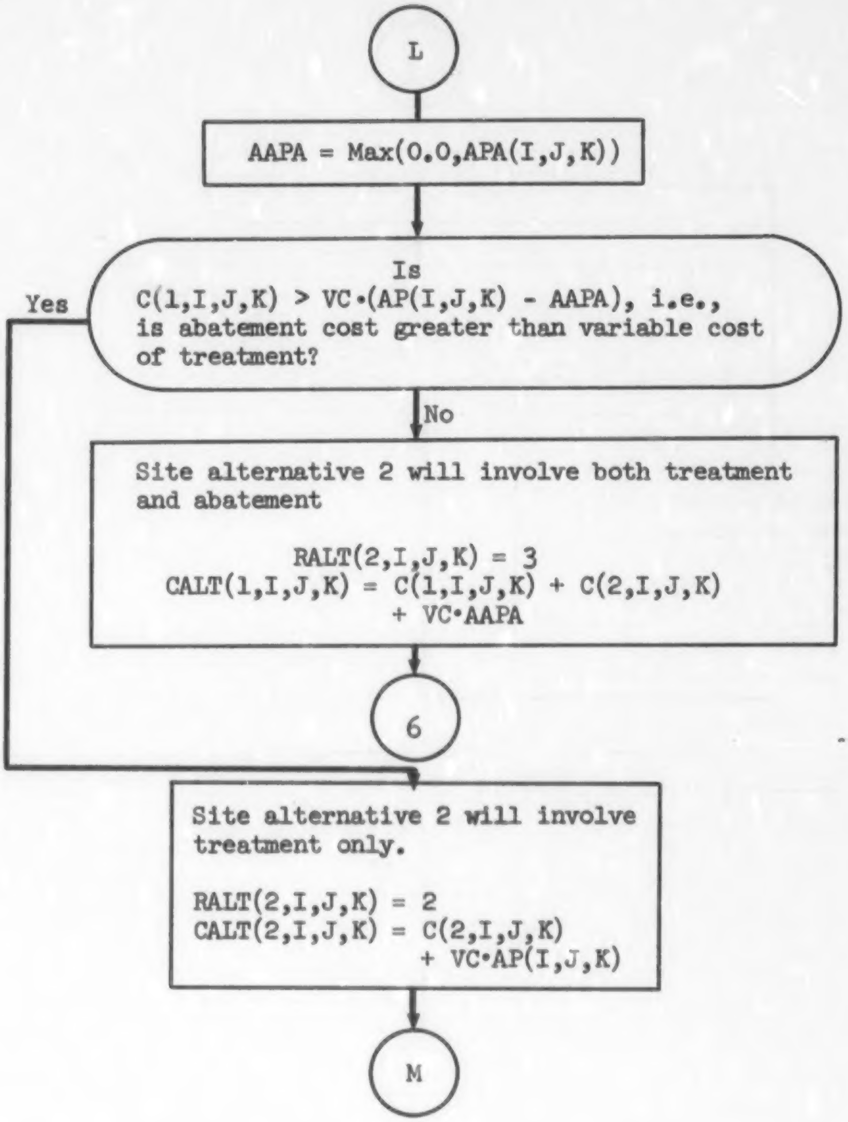

L
AAPA = Max(0.0,APA(I,J,K))
Is
C(1,I,J,K) > VC·(AP(I,J,K) - AAPA), i.e., is abatement cost greater than variable cost of treatment?
Yes
No
Site alternative 2 will involve both treatment and abatement
RALT(2,I,J,K) = 3
CALT(1,I,J,K) = C(1,I,J,K) + C(2,I,J,K) + VC·AAPA
6
Site alternative 2 will involve treatment only.
RALT(2,I,J,K) = 2
CALT(2,I,J,K) = C(2,I,J,K) + VC·AP(I,J,K)
M

Figure C.14 Program ALCOT

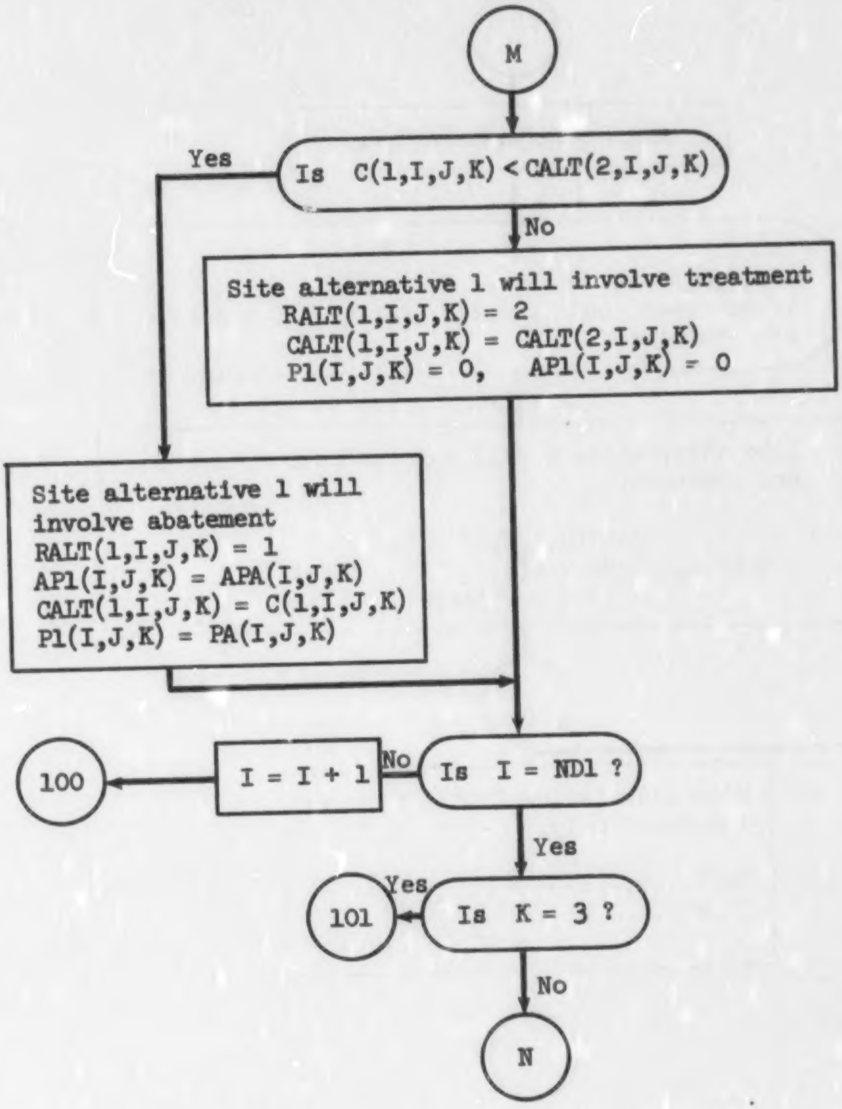

M
Is C(1,I,J,K) < CALT(2,I,J,K)
Yes
No
Site alternative 1 will involve treatment
RALT(1,I,J,K) = 2
CALT(1,I,J,K) = CALT(2,I,J,K)
P1(I,J,K) = 0, AP1(I,J,K) = 0
Site alternative 1 will involve abatement
RALT(1,I,J,K) = 1
AP1(I,J,K) = APA(I,J,K)
CALT(1,I,J,K) = C(1,I,J,K)
P1(I,J,K) = PA(I,J,K)
Is I = ND1 ?
No
I = I + 1
100
Yes
Is K = 3 ?
Yes
101
No
N

Figure C.14 Program ALCOT

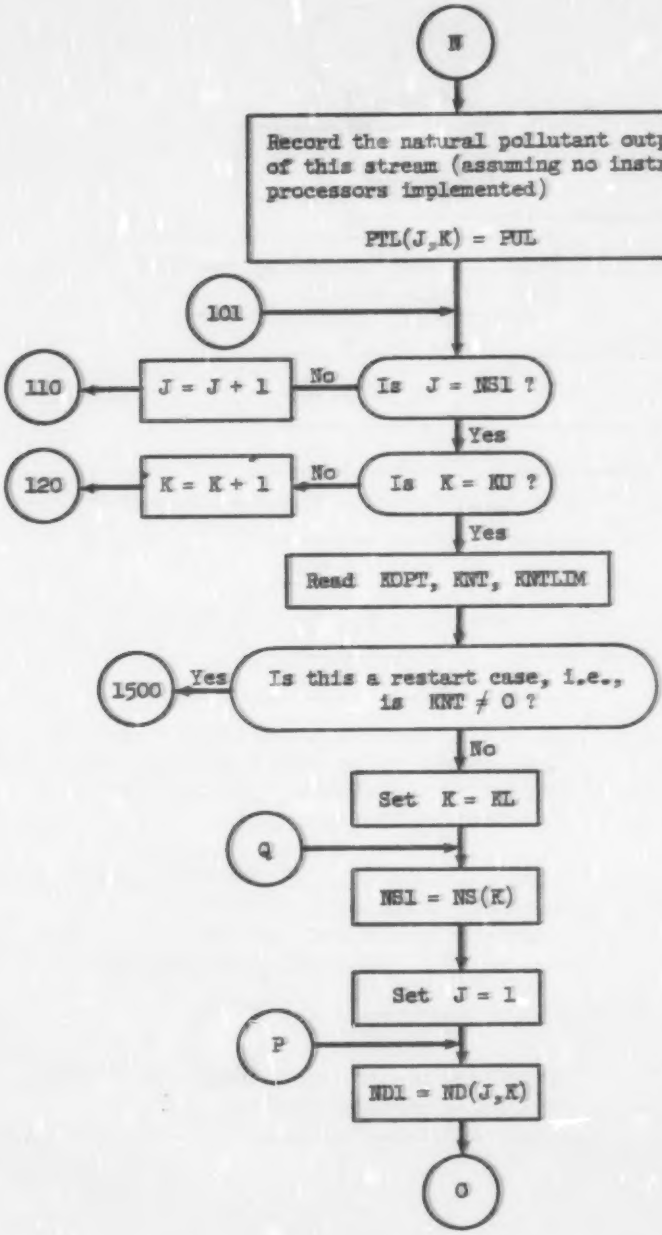
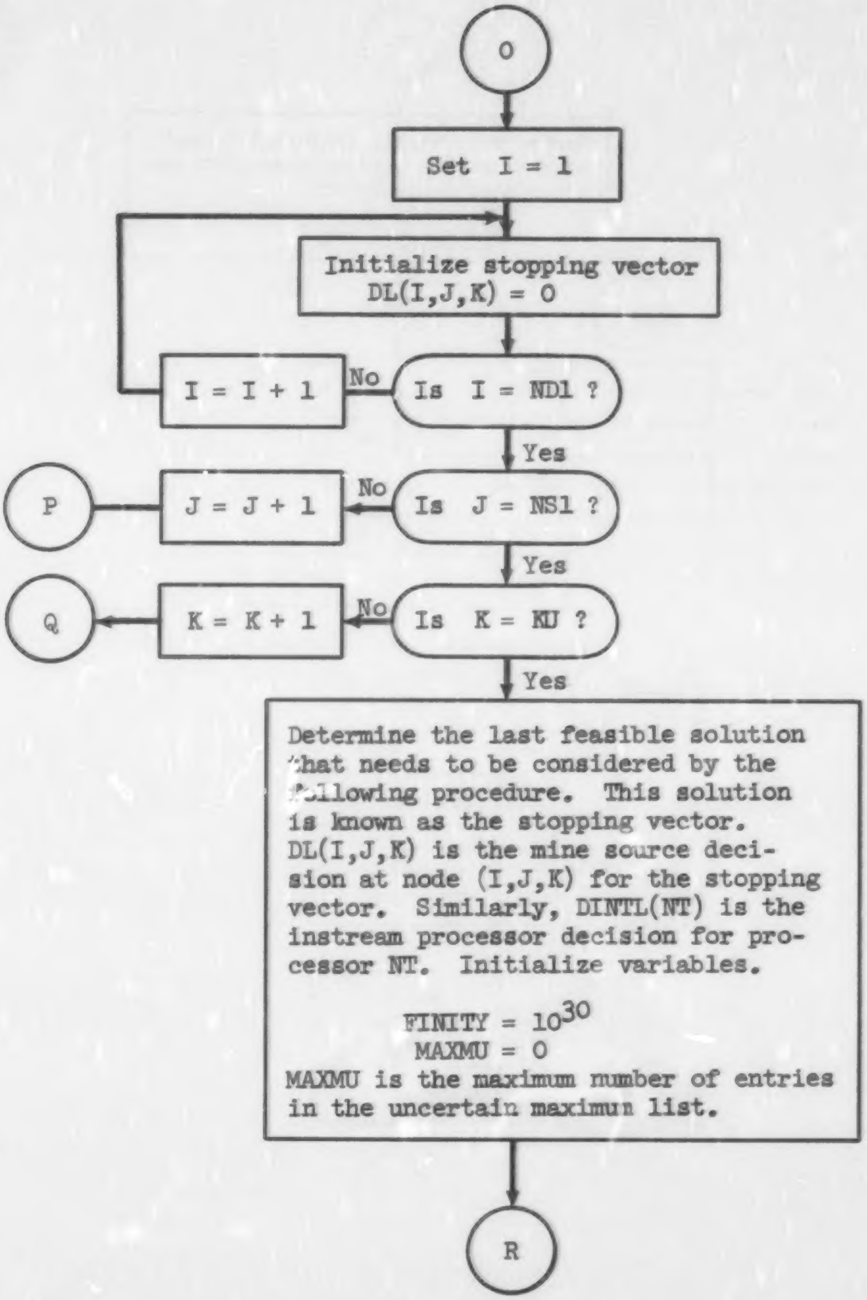

N
Record the natural pollutant output of this stream (assuming no instream processors implemented)
PTL(J,K) = PUL
101
110
J = J + 1
No
Is J = NS1 ?
Yes
120
K = K + 1
No
Is K = KU ?
Yes
Read KOPT, KNT, KNTLIM
1500
Yes
Is this a restart case, i.e., is KNT ≠ 0 ?
No
Set K = KL
Q
NS1 = NS(K)
Set J = 1
P
ND1 = ND(J,K)
O

Figure C.14 Program ALCOT



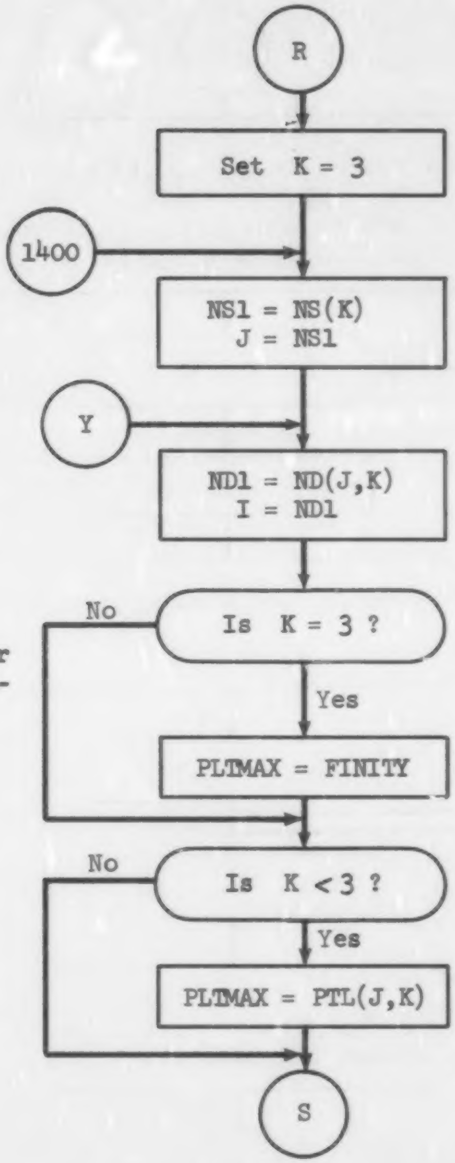
O
Set I = 1
Initialize stopping vector
DL(I,J,K) = 0
Is I = ND1 ?
No
I = I + 1
Yes
Is J = NS1 ?
No
J = J + 1
P
Yes
Is K = KU ?
No
K = K + 1
Q
Yes
R

Determine the last feasible solution that needs to be considered by the following procedure. This solution is known as the stopping vector. DL(I,J,K) is the mine source decision at node (I,J,K) for the stopping vector. Similarly, DINTL(NT) is the instream processor decision for processor NT. Initialize variables.

FINITY = 10^{30}
MAXMU = 0

MAXMU is the maximum number of entries in the uncertain maximum list.

Figure C.14 Program ALCOT


R
Set K = 3
1400
NS1 = NS(K)
J = NS1
Y
ND1 = ND(J,K)
I = ND1
No
Is K = 3 ?
Yes
PLTMAX = FINITY
No
Is K < 3 ?
Yes
PLTMAX = PTL(J,K)
S

PLTMAX is the allowable upper limit on pollution flow.

Figure C.14 Program ALCOT

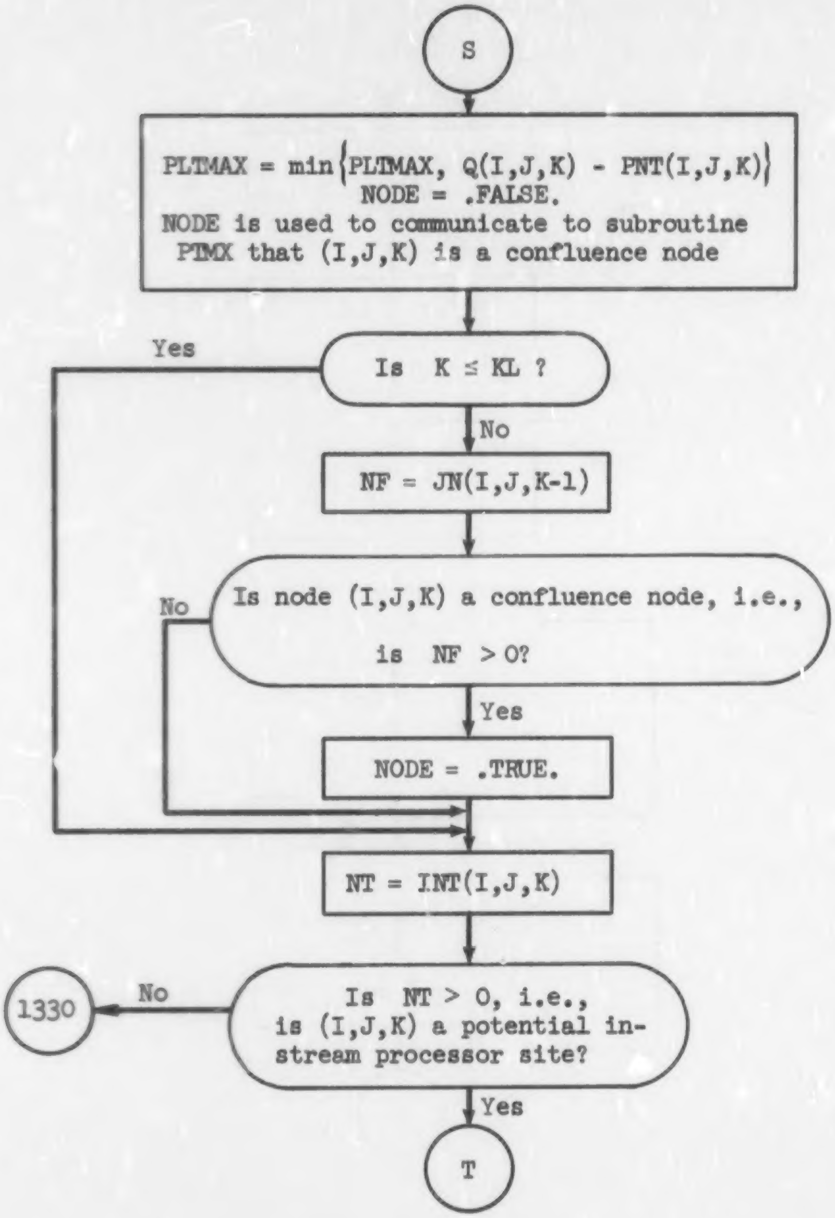

S
PLTMAX = min{PLTMAX, Q(I,J,K) - PNT(I,J,K)}
NODE = .FALSE.
NODE is used to communicate to subroutine PTMX that (I,J,K) is a confluence node
Yes
Is K ≤ KL ?
No
NF = JN(I,J,K-1)
No
Is node (I,J,K) a confluence node, i.e., is NF > 0?
Yes
NODE = .TRUE.
NT = INT(I,J,K)
1330
No
Is NT > 0, i.e., is (I,J,K) a potential in-stream processor site?
Yes
T

Figure C.14 Program ALCOT

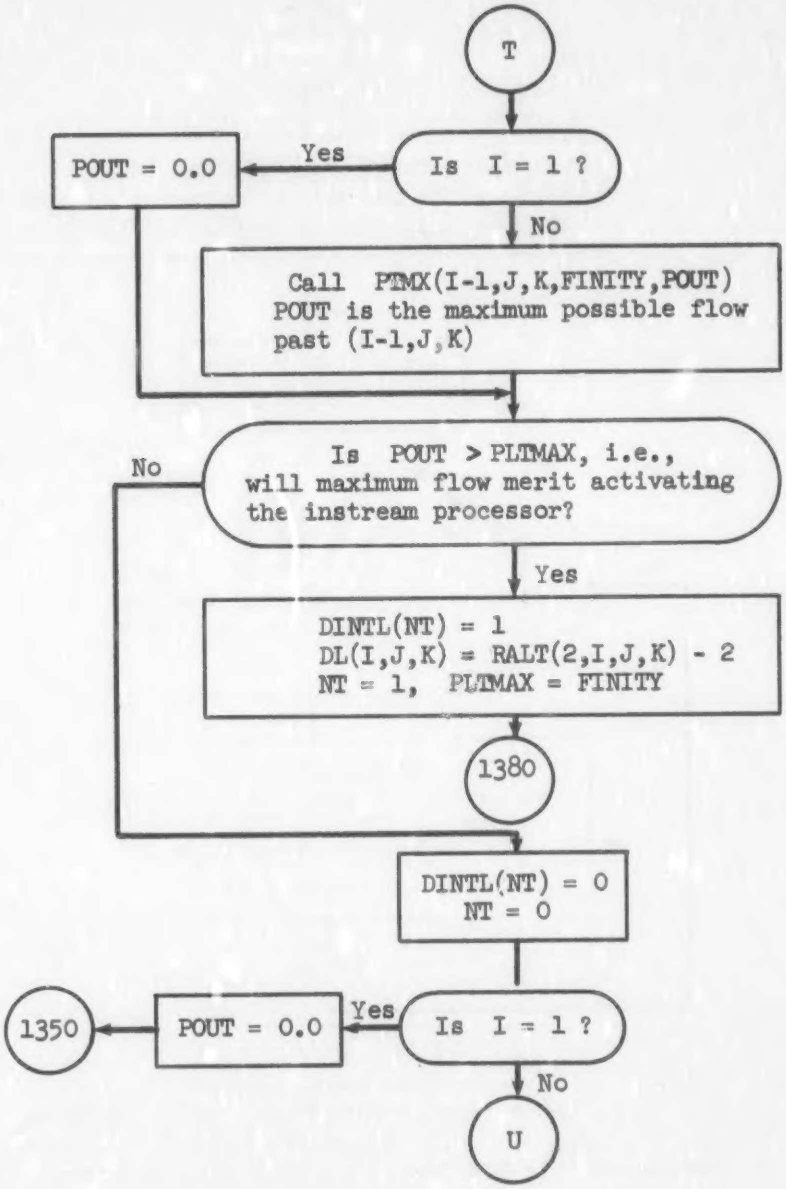

T
Is I = 1 ?
Yes
POUT = 0.0
No
Call PTMX(I-1,J,K,FINITY,POUT)
POUT is the maximum possible flow past (I-1,J,K)
Is POUT > PLTMAX, i.e., will maximum flow merit activating the instream processor?
No
Yes
DINTL(NT) = 1
DL(I,J,K) = RALT(2,I,J,K) - 2
NT = 1, PLTMAX = FINITY
1380
DINTL(NT) = 0
NT = 0
Is I = 1 ?
Yes
POUT = 0.0
1350
No
U

Figure C.14 Program ALCOT

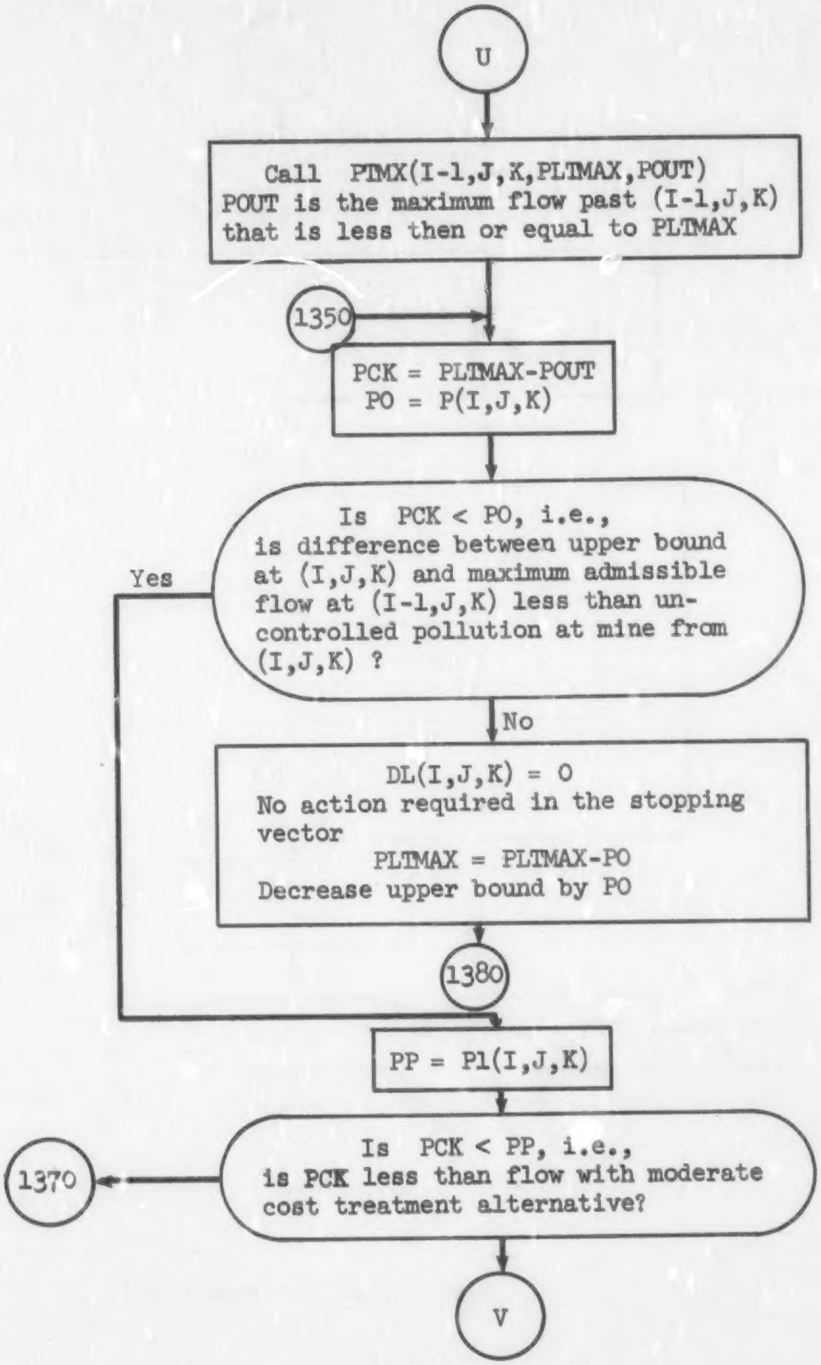

U
Call PTMX(I-1,J,K,PLTMAX,POUT)
POUT is the maximum flow past (I-1,J,K) that is less then or equal to PLTMAX
1350
PCK = PLTMAX-POUT
PO = P(I,J,K)
Is PCK < PO, i.e., is difference between upper bound at (I,J,K) and maximum admissible flow at (I-1,J,K) less than uncontrolled pollution at mine from (I,J,K) ?
Yes
No
DL(I,J,K) = 0
No action required in the stopping vector
PLTMAX = PLTMAX-PO
Decrease upper bound by PO
1380
PP = P1(I,J,K)
Is PCK < PP, i.e., is PCK less than flow with moderate cost treatment alternative?
1370
V

Figure C.14 Program ALCOT

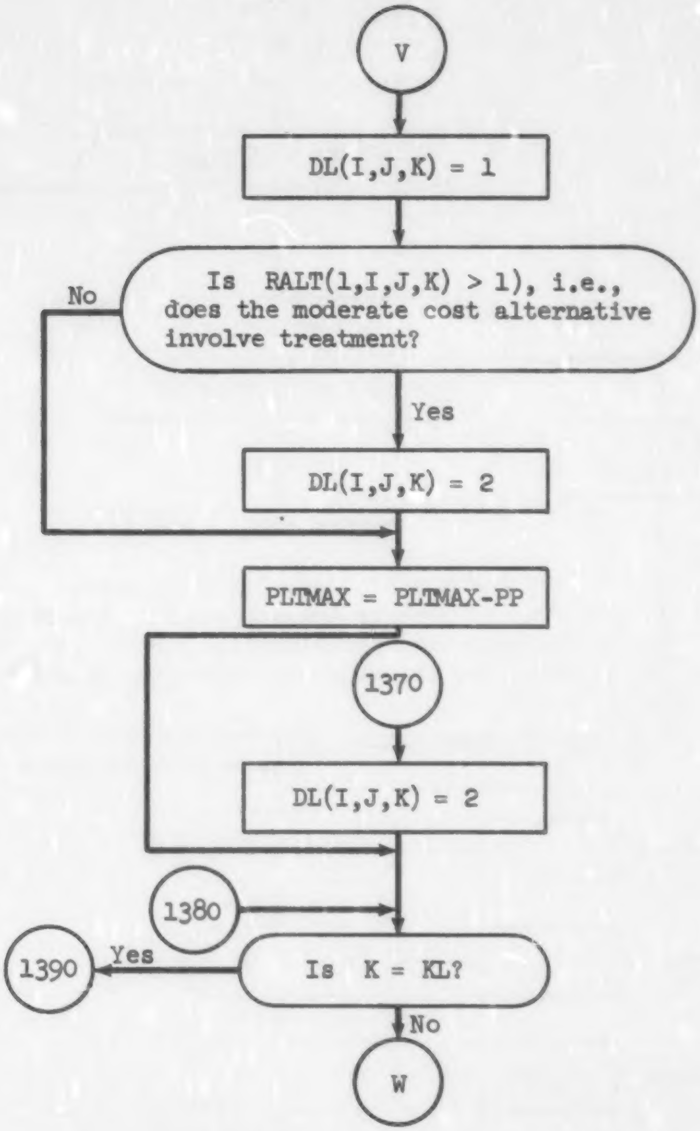

V
DL(I,J,K) = 1
Is RALT(1,I,J,K) > 1), i.e., does the moderate cost alternative involve treatment?
No
Yes
DL(I,J,K) = 2
PLTMAX = PLTMAX-PP
1370
DL(I,J,K) = 2
1380
1390
Yes
Is K = KL?
No
W

Figure C.14 Program ALCOT

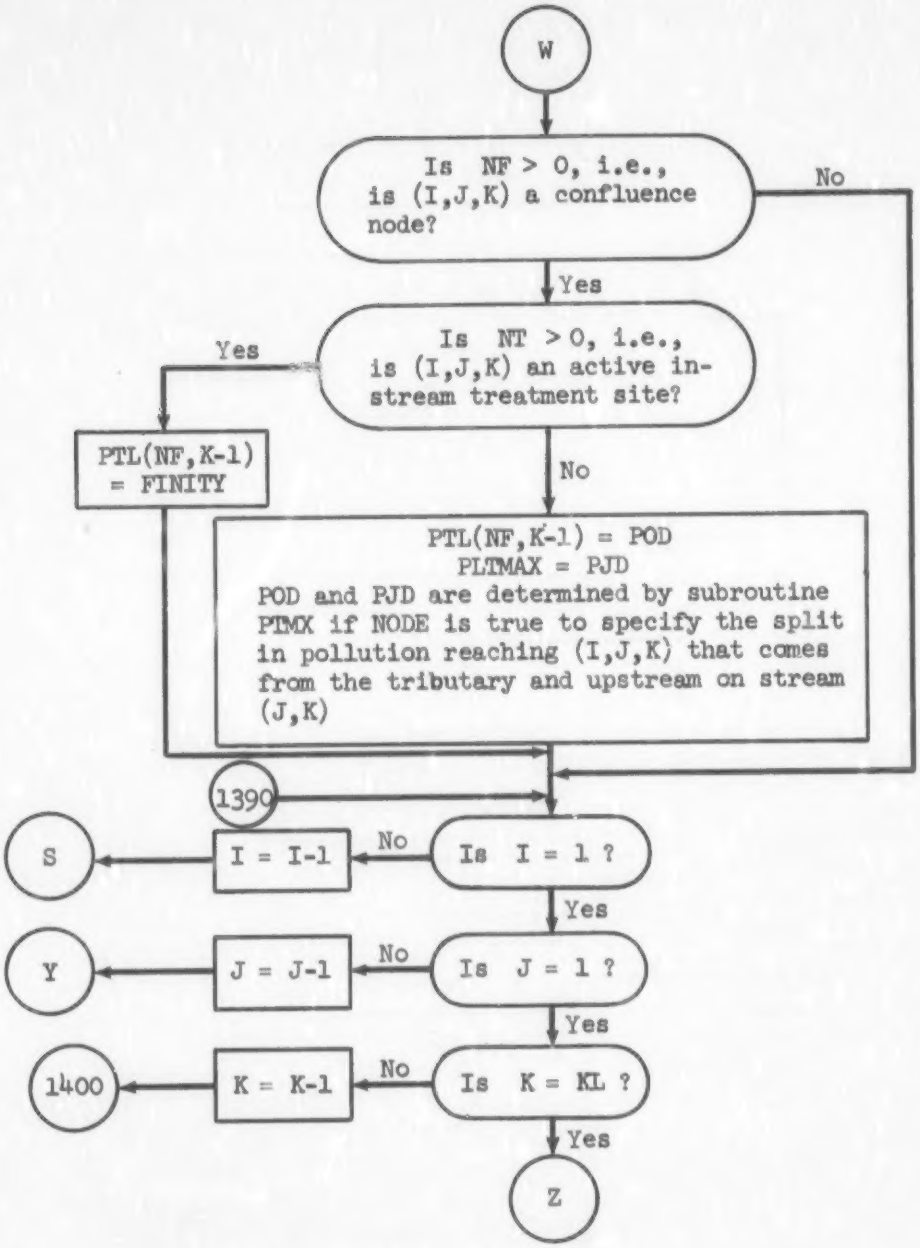

W
Is NF > 0, i.e., is (I,J,K) a confluence node?
No
Yes
Is NT > 0, i.e., is (I,J,K) an active in-stream treatment site?
Yes
PTL(NF,K-1) = FINITY
No
PTL(NF,K-1) = POD
PLTMAX = PJD
POD and PJD are determined by subroutine PTMX if NODE is true to specify the split in pollution reaching (I,J,K) that comes from the tributary and upstream on stream (J,K)
1390
S
I = I-1
No
Is I = 1 ?
Yes
Y
J = J-1
No
Is J = 1 ?
Yes
1400
K = K-1
No
Is K = KL ?
Yes
Z

Figure C.14 Program ALCOT

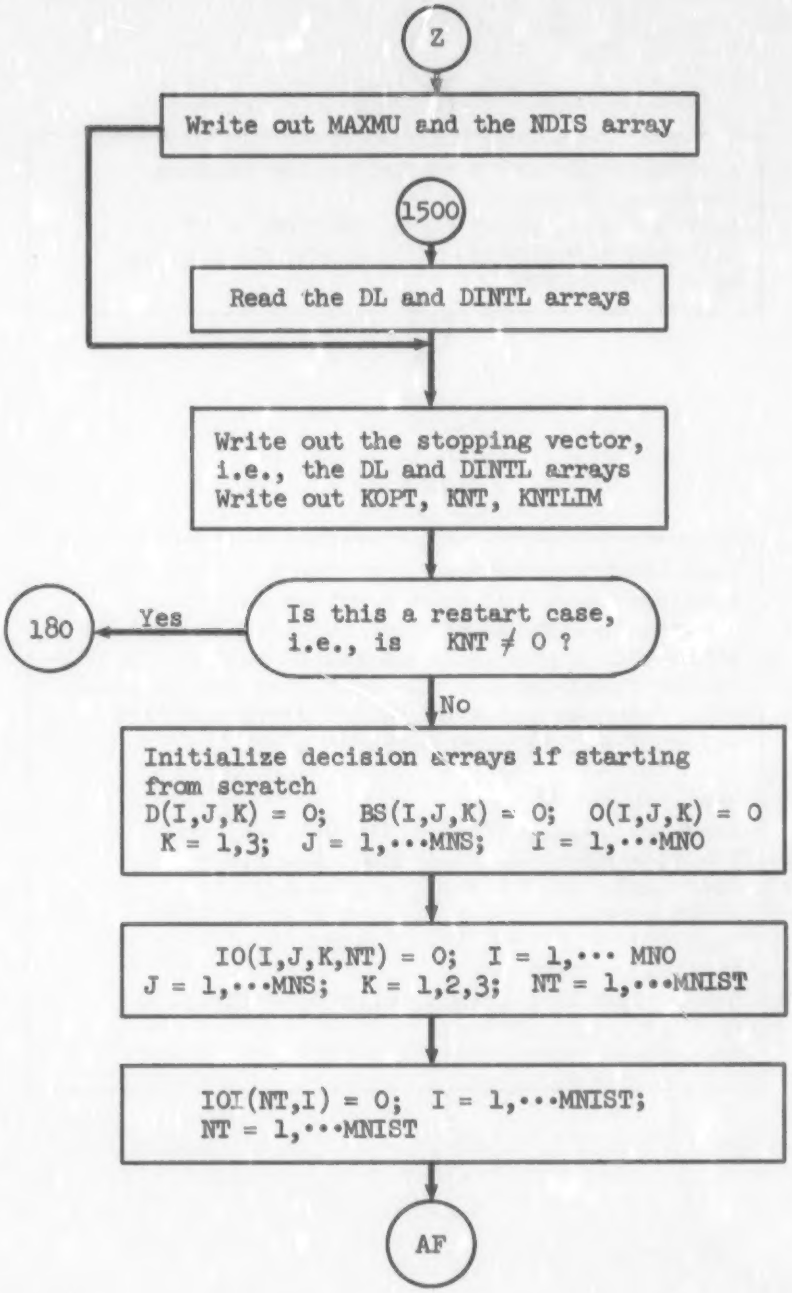
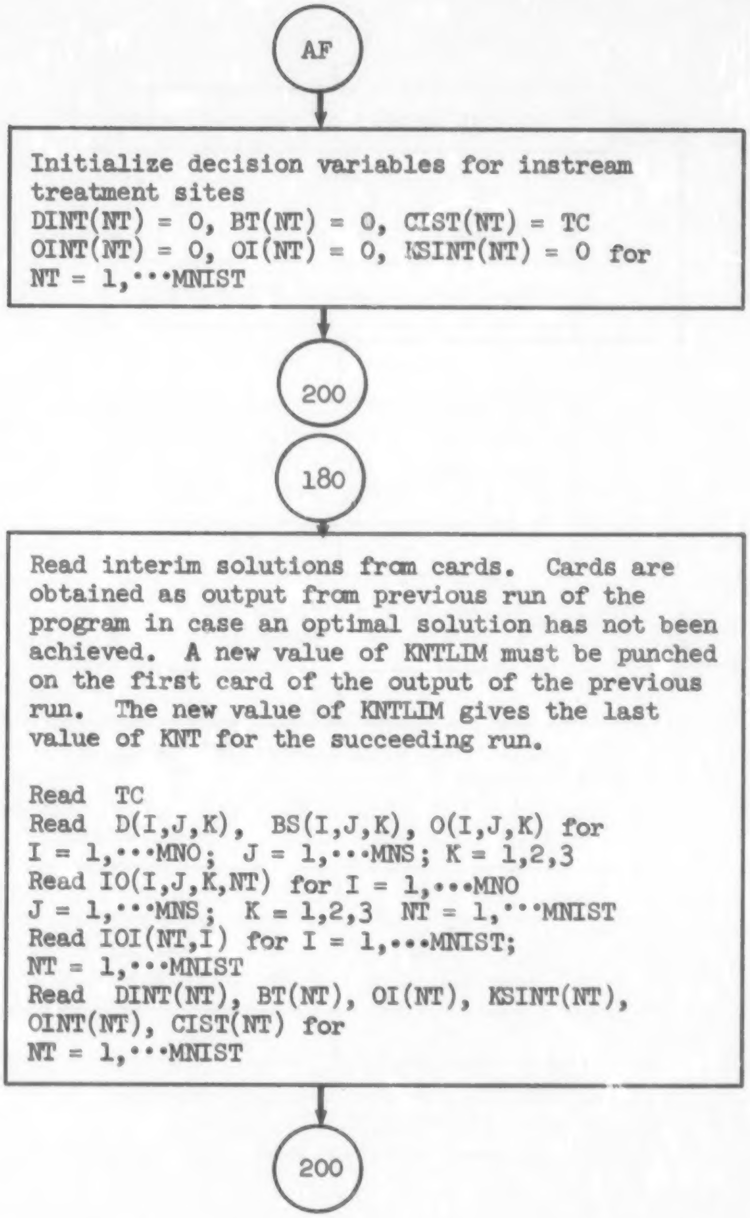

Z
Write out MAXMU and the NDIS array
1500
Read the DL and DINTL arrays
Write out the stopping vector,
i.e., the DL and DINTL arrays
Write out KOPT, KNT, KNTLIM
Is this a restart case,
i.e., is KNT ≠ 0 ?
Yes
180
No
Initialize decision arrays if starting
from scratch
D(I,J,K) = 0; BS(I,J,K) = 0; O(I,J,K) = 0
K = 1,3; J = 1,···MNS; I = 1,···MNO
IO(I,J,K,NT) = 0; I = 1,··· MNO
J = 1,···MNS; K = 1,2,3; NT = 1,···MNIST
IOI(NT,I) = 0; I = 1,···MNIST;
NT = 1,···MNIST
AF

Figure C.14 Program ALCOT



Initialize decision variables for instream treatment sites
DINT(NT) = 0, BT(NT) = 0, CIST(NT) = TC
OINT(NT) = 0, OI(NT) = 0, KSINT(NT) = 0 for
NT = 1,···MNIST

Read interim solutions from cards. Cards are obtained as output from previous run of the program in case an optimal solution has not been achieved. A new value of KNTLIM must be punched on the first card of the output of the previous run. The new value of KNTLIM gives the last value of KNT for the succeeding run.

Read TC
Read D(I,J,K), BS(I,J,K), O(I,J,K) for
I = 1,···MNO; J = 1,···MNS; K = 1,2,3
Read IO(I,J,K,NT) for I = 1,···MNO
J = 1,···MNS; K = 1,2,3 NT = 1,···MNIST
Read IOI(NT,I) for I = 1,···MNIST;
NT = 1,···MNIST
Read DINT(NT), BT(NT), OI(NT), KSINT(NT),
OINT(NT), CIST(NT) for
NT = 1,···MNIST

Figure C.14 Program ALCOT

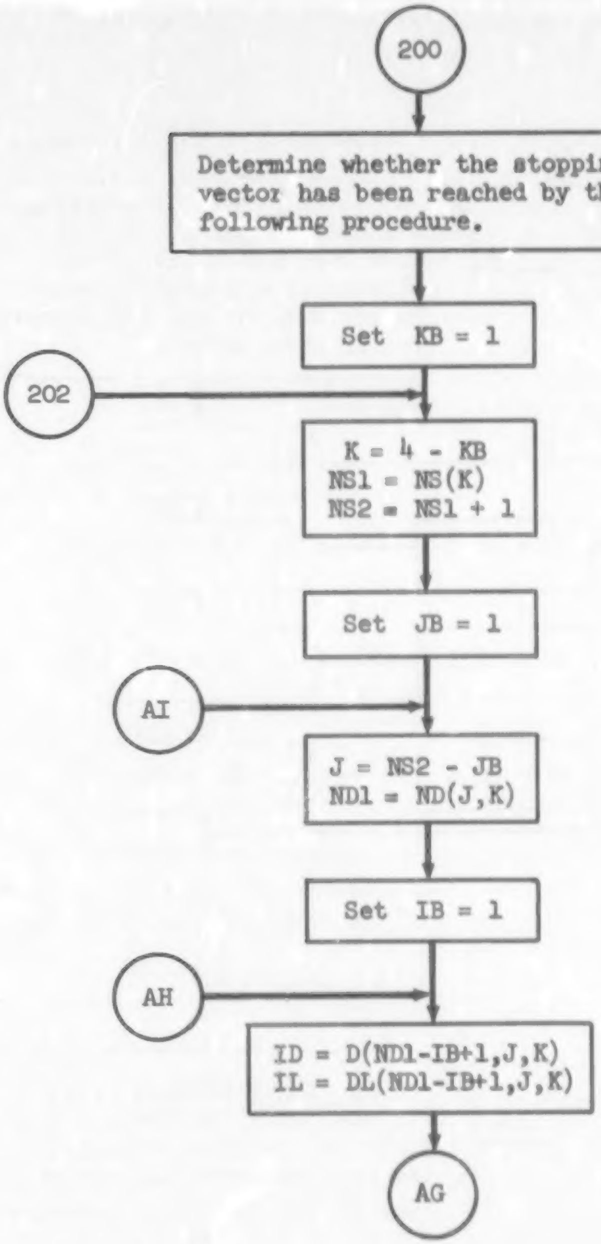

200
Determine whether the stopping vector has been reached by the following procedure.
Set KB = 1
202
K = 4 - KB
NS1 = NS(K)
NS2 = NS1 + 1
Set JB = 1
AI
J = NS2 - JB
ND1 = ND(J,K)
Set IB = 1
AH
ID = D(ND1-IB+1,J,K)
IL = DL(ND1-IB+1,J,K)
AG

Figure C.14 Program ALCOT

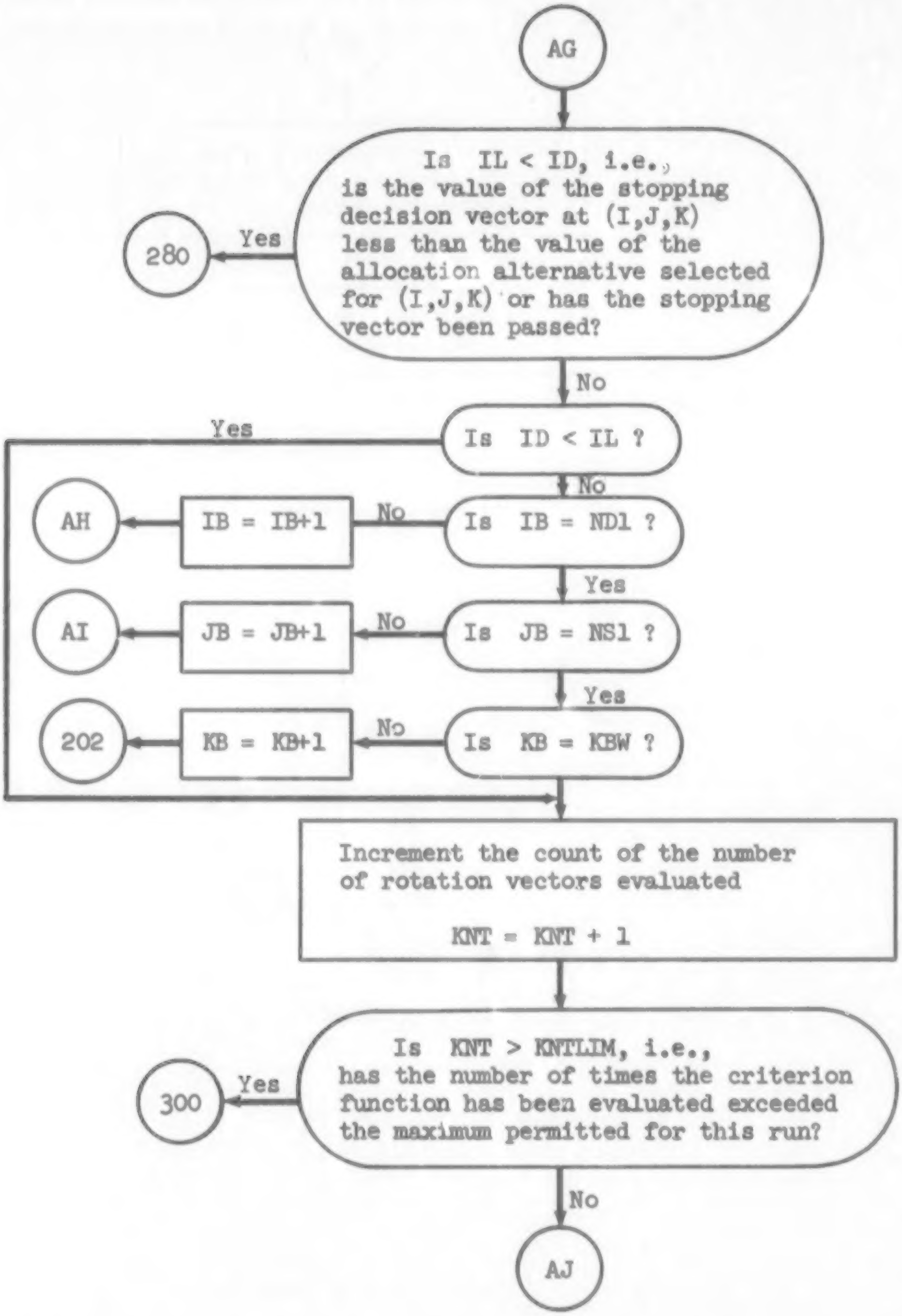

AG
Is IL < ID, i.e., is the value of the stopping decision vector at (I,J,K) less than the value of the allocation alternative selected for (I,J,K) or has the stopping vector been passed?
Yes
280
No
Is ID < IL ?
Yes
No
Is IB = ND1 ?
No
IB = IB+1
AH
Yes
Is JB = NS1 ?
No
JB = JB+1
AI
Yes
Is KB = KBW ?
No
KB = KB+1
202
Increment the count of the number of rotation vectors evaluated
KNT = KNT + 1
Is KNT > KNTLIM, i.e., has the number of times the criterion function has been evaluated exceeded the maximum permitted for this run?
Yes
300
No
AJ

Figure C.14 Program ALCOT

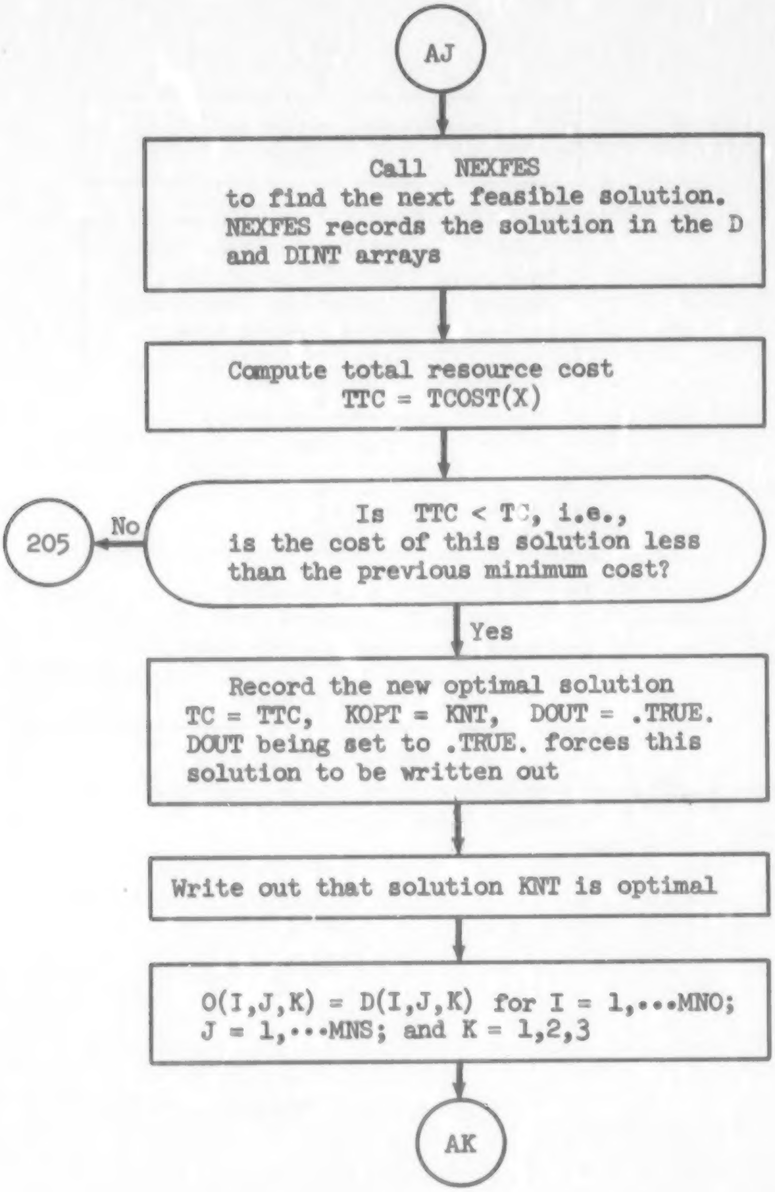

AJ
Call NEXFES
to find the next feasible solution.
NEXFES records the solution in the D
and DINT arrays
Compute total resource cost
TTC = TCOST(X)
Is TTC < TC, i.e.,
is the cost of this solution less
than the previous minimum cost?
No
205
Yes
Record the new optimal solution
TC = TTC, KOPT = KNT, DOUT = .TRUE.
DOUT being set to .TRUE. forces this
solution to be written out
Write out that solution KNT is optimal
O(I,J,K) = D(I,J,K) for I = 1,...MNO;
J = 1,...MNS; and K = 1,2,3
AK

Figure C.14 Program ALCOT

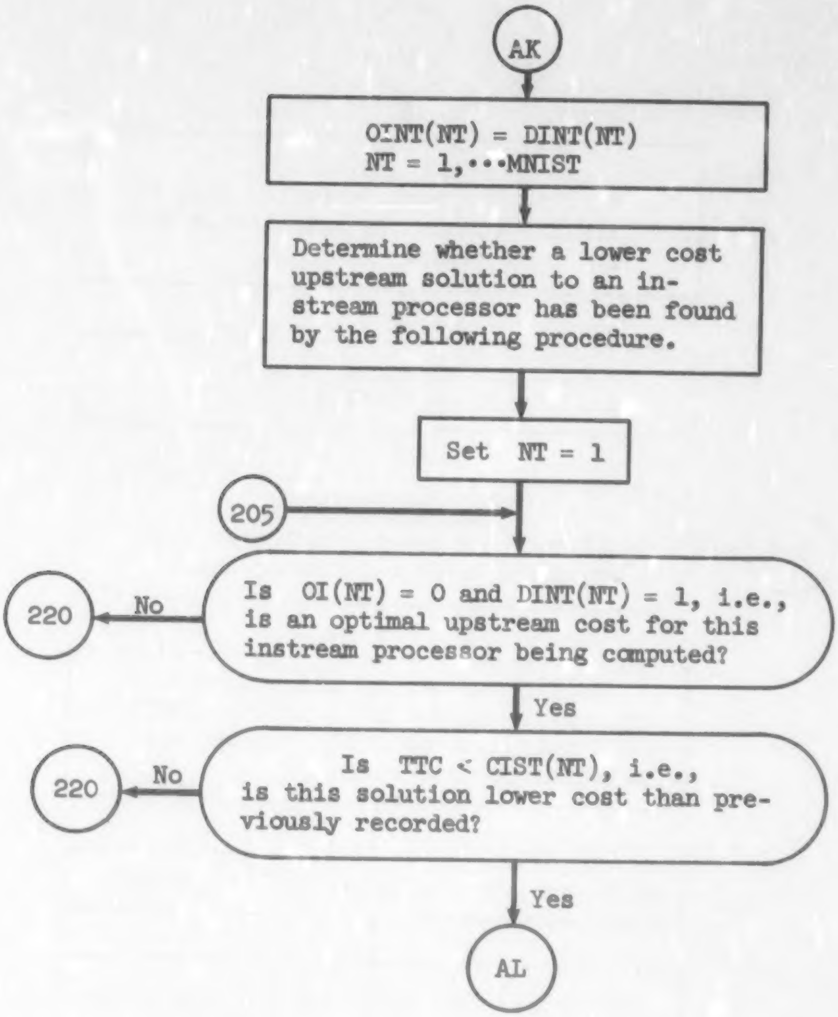

AK
OINT(NT) = DINT(NT)
NT = 1,•••MNIST
Determine whether a lower cost upstream solution to an instream processor has been found by the following procedure.
Set NT = 1
205
Is OI(NT) = 0 and DINT(NT) = 1, i.e., is an optimal upstream cost for this instream processor being computed?
No
220
Yes
Is TTC < CIST(NT), i.e., is this solution lower cost than previously recorded?
No
220
Yes
AL

Figure C.14 Program ALCOT

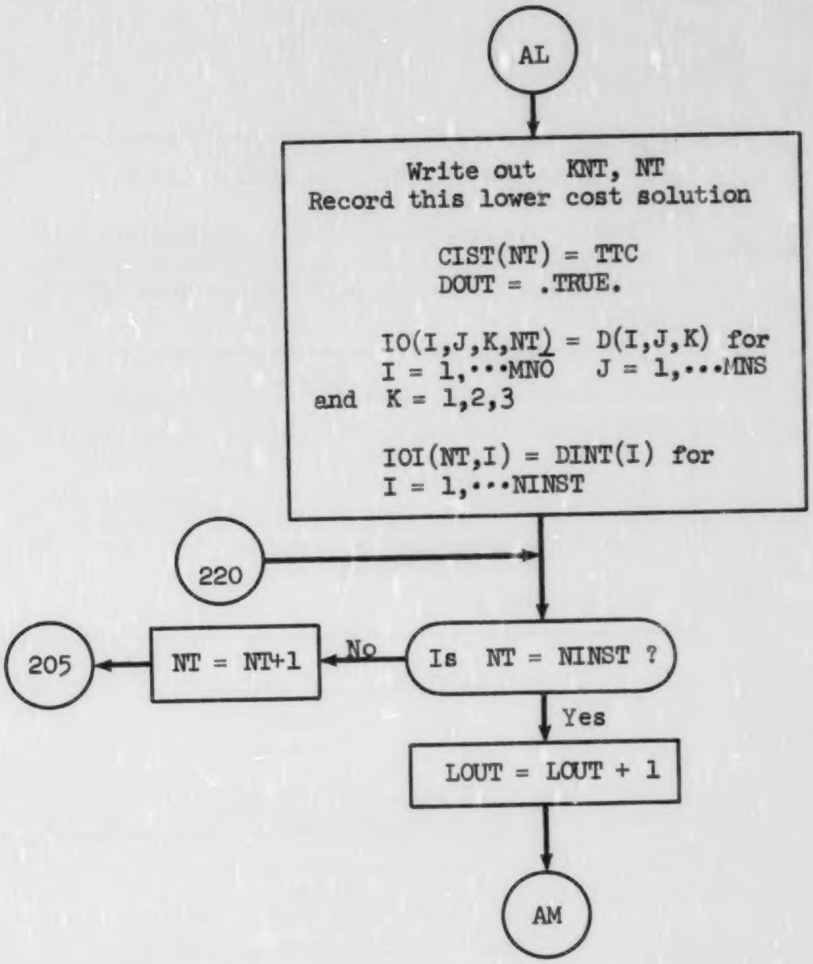
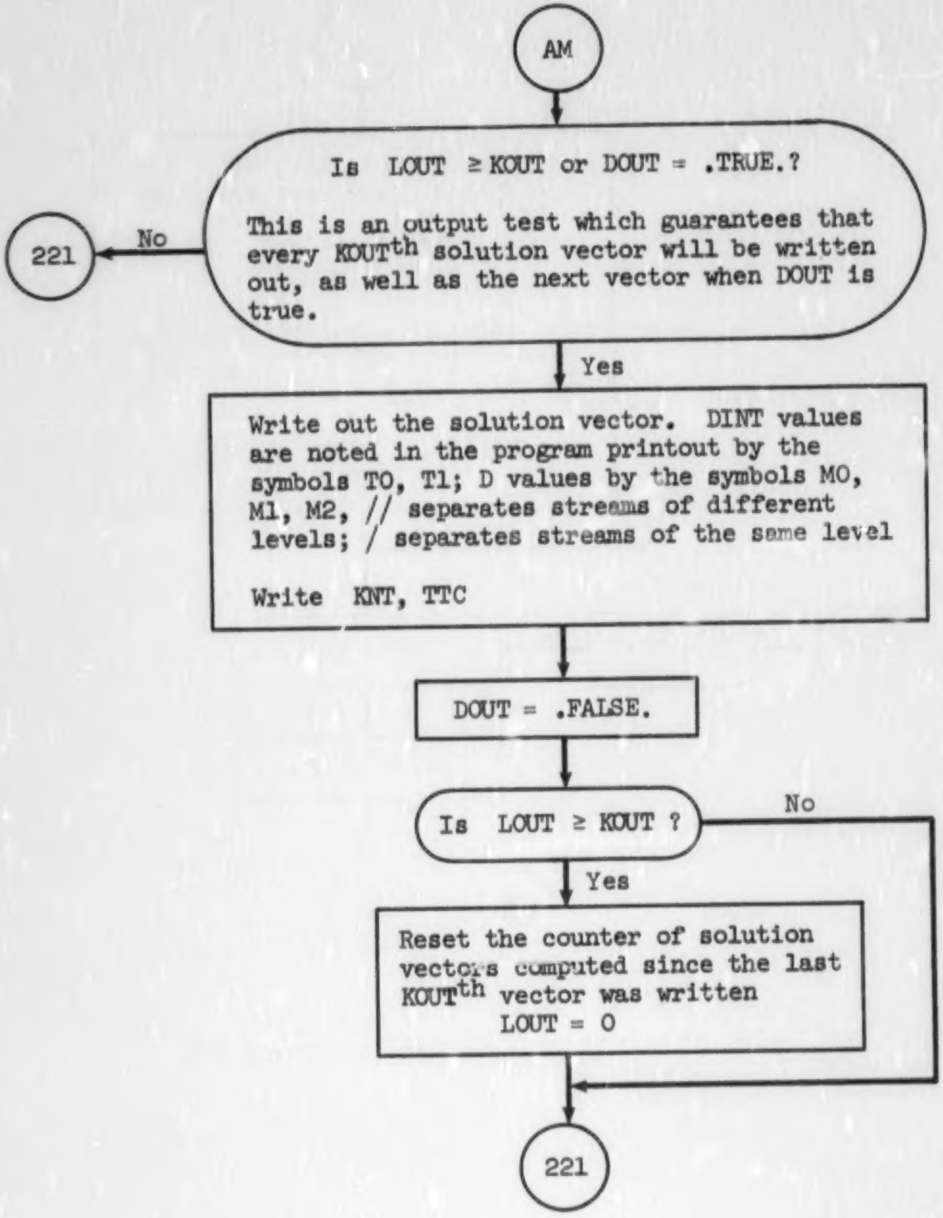

AL
Write out KNT, NT
Record this lower cost solution
CIST(NT) = TTC
DOUT = .TRUE.
IO(I,J,K,NT) = D(I,J,K) for
I = 1,•••MNO J = 1,•••MNS
and K = 1,2,3
IOI(NT,I) = DINT(I) for
I = 1,•••NINST
220
Is NT = NINST ?
No
NT = NT+1
205
Yes
LOUT = LOUT + 1
AM

Figure C.14 Program ALCOT


AM
Is LOUT ≥ KOUT or DOUT = .TRUE.?
No
221
Yes
DOUT = .FALSE.
Is LOUT ≥ KOUT ?
No
Yes
LOUT = 0
221

This is an output test which guarantees that every KOUTth solution vector will be written out, as well as the next vector when DOUT is true.

Write out the solution vector. DINT values are noted in the program printout by the symbols T0, T1; D values by the symbols M0, M1, M2, // separates streams of different levels; / separates streams of the same level

Write KNT, TTC

Reset the counter of solution vectors computed since the last KOUTth vector was written

Figure C.14 Program ALCOT

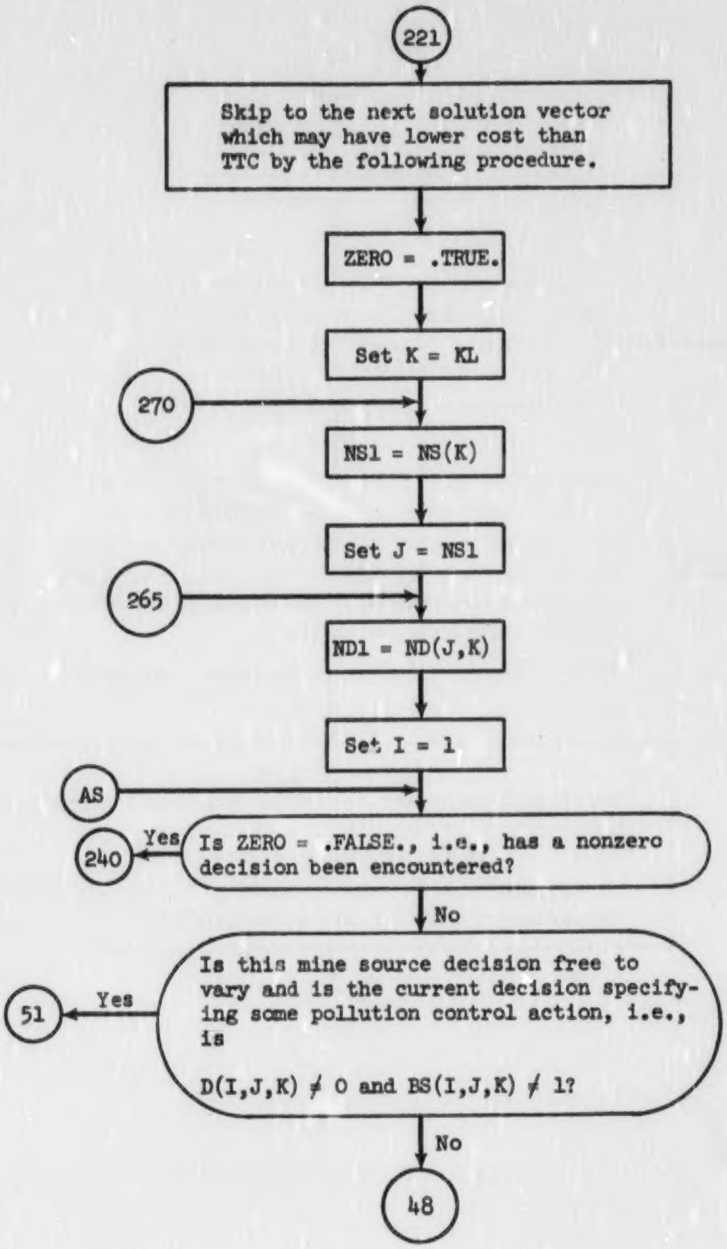

221
Skip to the next solution vector which may have lower cost than TTC by the following procedure.
ZERO = .TRUE.
Set K = KL
270
NS1 = NS(K)
Set J = NS1
265
ND1 = ND(J,K)
Set I = 1
AS
Yes
240
Is ZERO = .FALSE., i.e., has a nonzero decision been encountered?
No
Is this mine source decision free to vary and is the current decision specifying some pollution control action, i.e., is
D(I,J,K) ≠ 0 and BS(I,J,K) ≠ 1?
Yes
51
No
48

Figure C.14 Program ALCOT

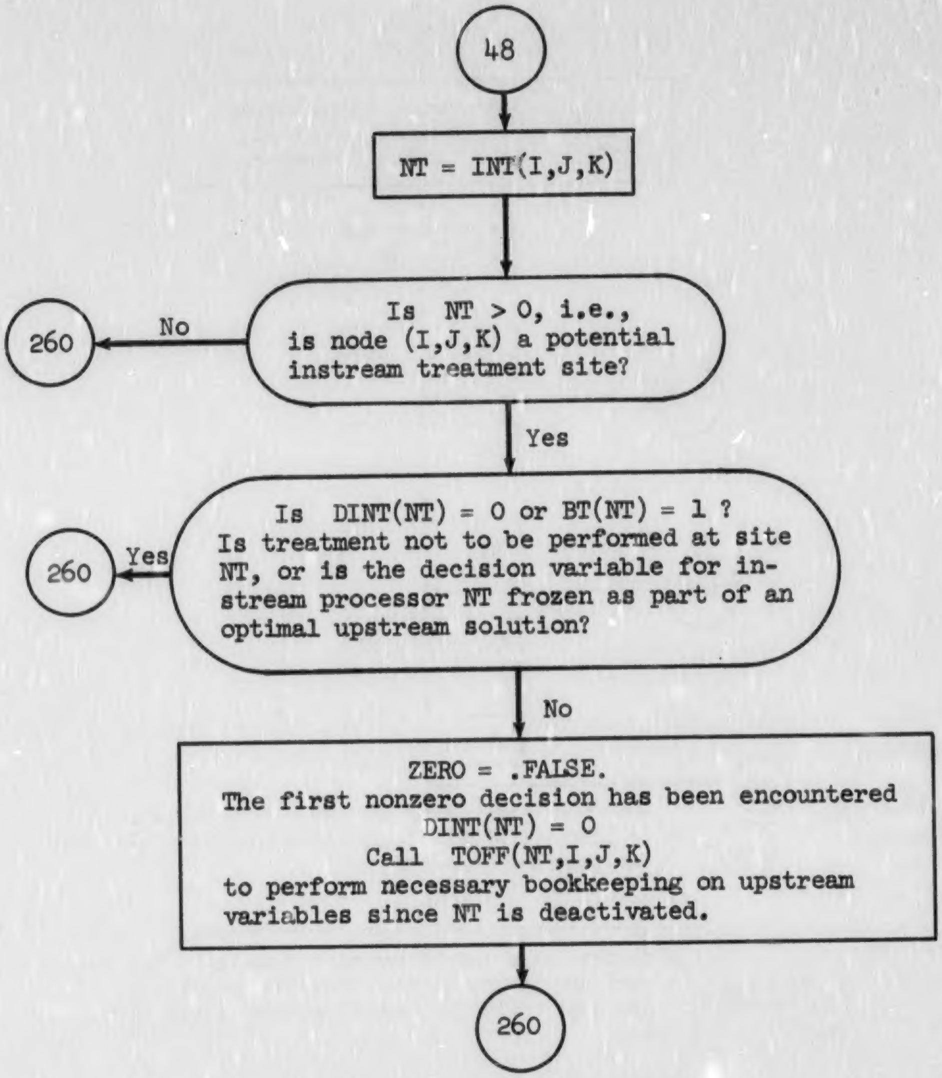

48
NT = INT(I,J,K)
Is NT > 0, i.e., is node (I,J,K) a potential instream treatment site?
No
260
Yes
Is DINT(NT) = 0 or BT(NT) = 1 ? Is treatment not to be performed at site NT, or is the decision variable for instream processor NT frozen as part of an optimal upstream solution?
Yes
260
No
ZERO = .FALSE.
The first nonzero decision has been encountered
DINT(NT) = 0
Call TOFF(NT,I,J,K)
to perform necessary bookkeeping on upstream variables since NT is deactivated.
260

Figure C.14 Program ALCOT

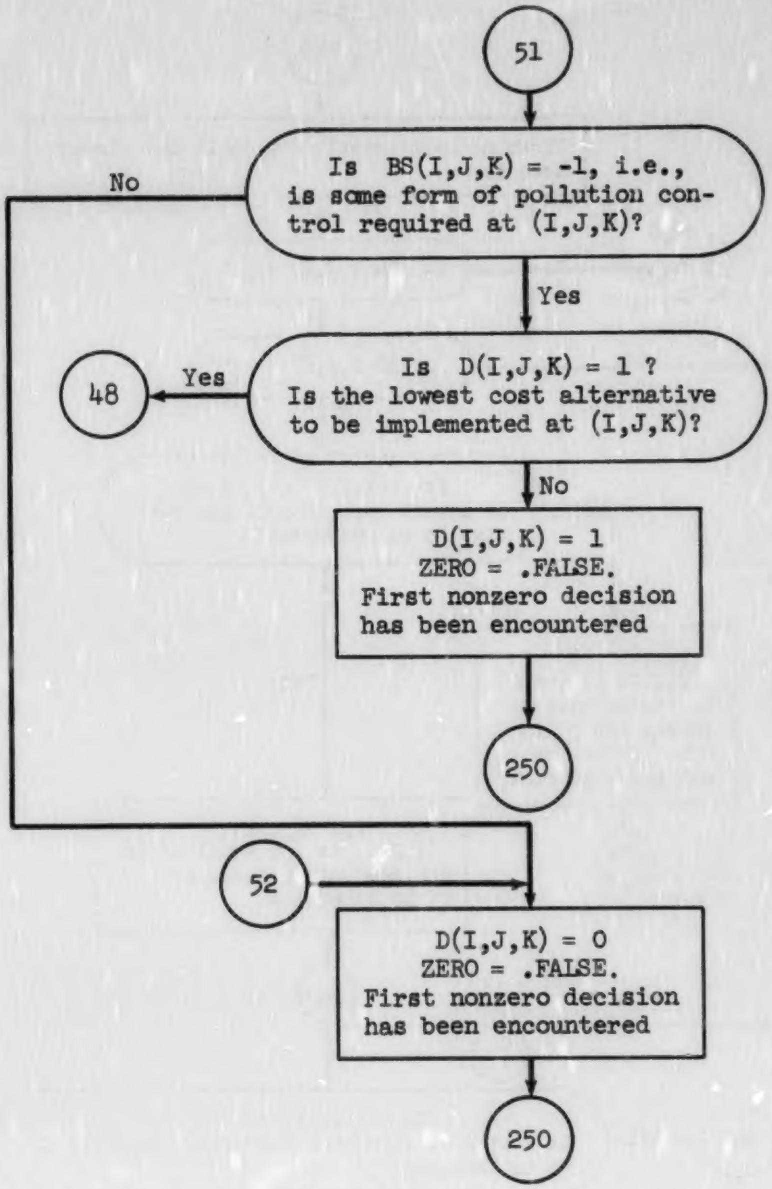

51
Is BS(I,J,K) = -1, i.e., is some form of pollution control required at (I,J,K)?
No
Yes
Is D(I,J,K) = 1 ? Is the lowest cost alternative to be implemented at (I,J,K)?
Yes
48
No
D(I,J,K) = 1
ZERO = .FALSE.
First nonzero decision has been encountered
250
52
D(I,J,K) = 0
ZERO = .FALSE.
First nonzero decision has been encountered
250

Figure C.14 Program ALCOT

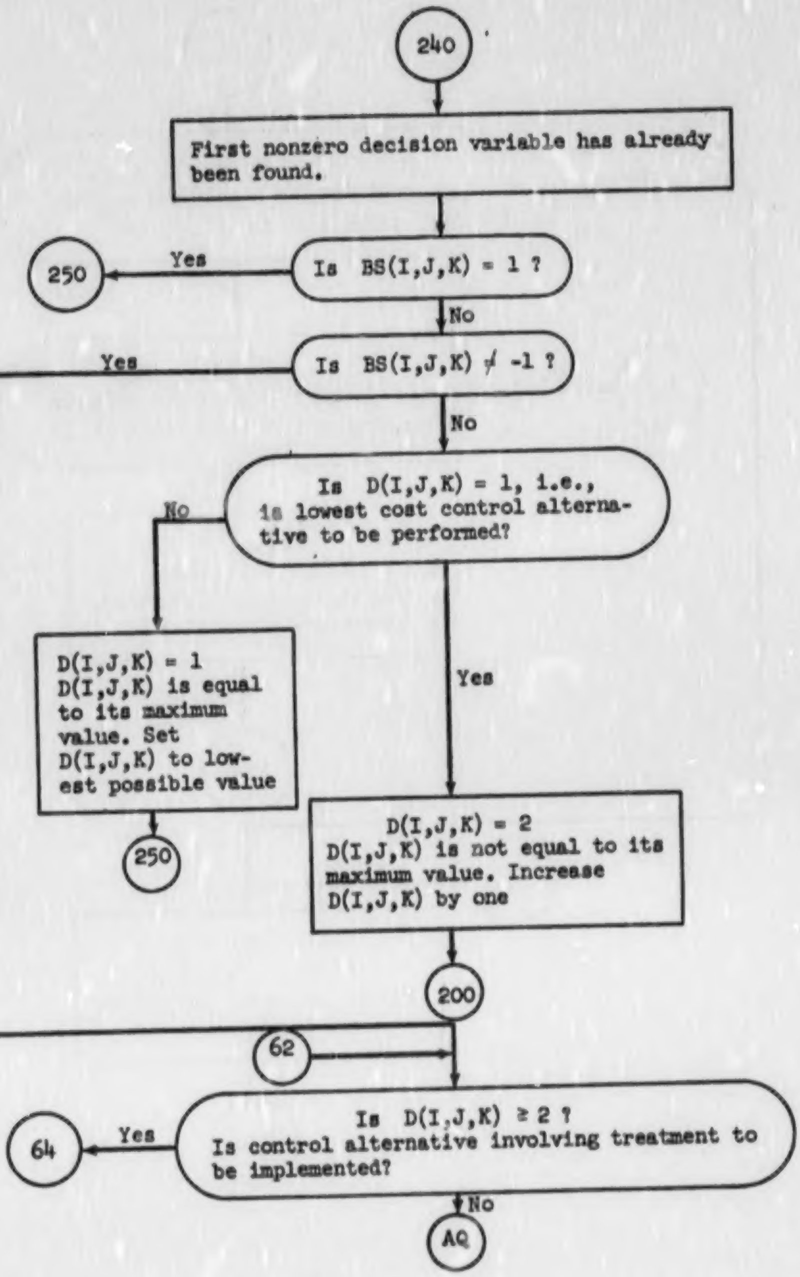

240
First nonzero decision variable has already been found.
Is BS(I,J,K) = 1 ?
Yes
250
No
Is BS(I,J,K) ≠ -1 ?
Yes
No
Is D(I,J,K) = 1, i.e., is lowest cost control alternative to be performed?
No
Yes
D(I,J,K) = 1
D(I,J,K) is equal to its maximum value. Set D(I,J,K) to lowest possible value
250
D(I,J,K) = 2
D(I,J,K) is not equal to its maximum value. Increase D(I,J,K) by one
200
62
Is D(I,J,K) ≥ 2 ?
Is control alternative involving treatment to be implemented?
Yes
64
No
AQ

Figure C.14 Program ALCOT

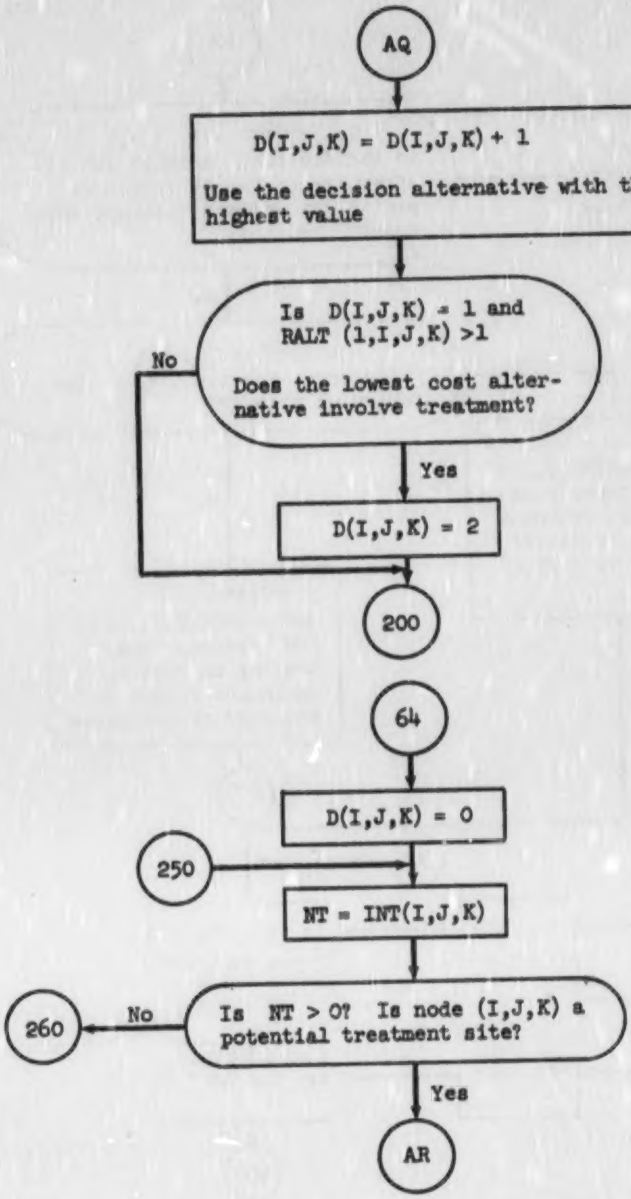

AQ
D(I,J,K) = D(I,J,K) + 1
Use the decision alternative with the next highest value
Is D(I,J,K) = 1 and RALT (1,I,J,K) >1
Does the lowest cost alternative involve treatment?
No
Yes
D(I,J,K) = 2
200
64
D(I,J,K) = 0
250
NT = INT(I,J,K)
Is NT > 0? Is node (I,J,K) a potential treatment site?
260
No
Yes
AR

Figure C.14 Program ALCOT

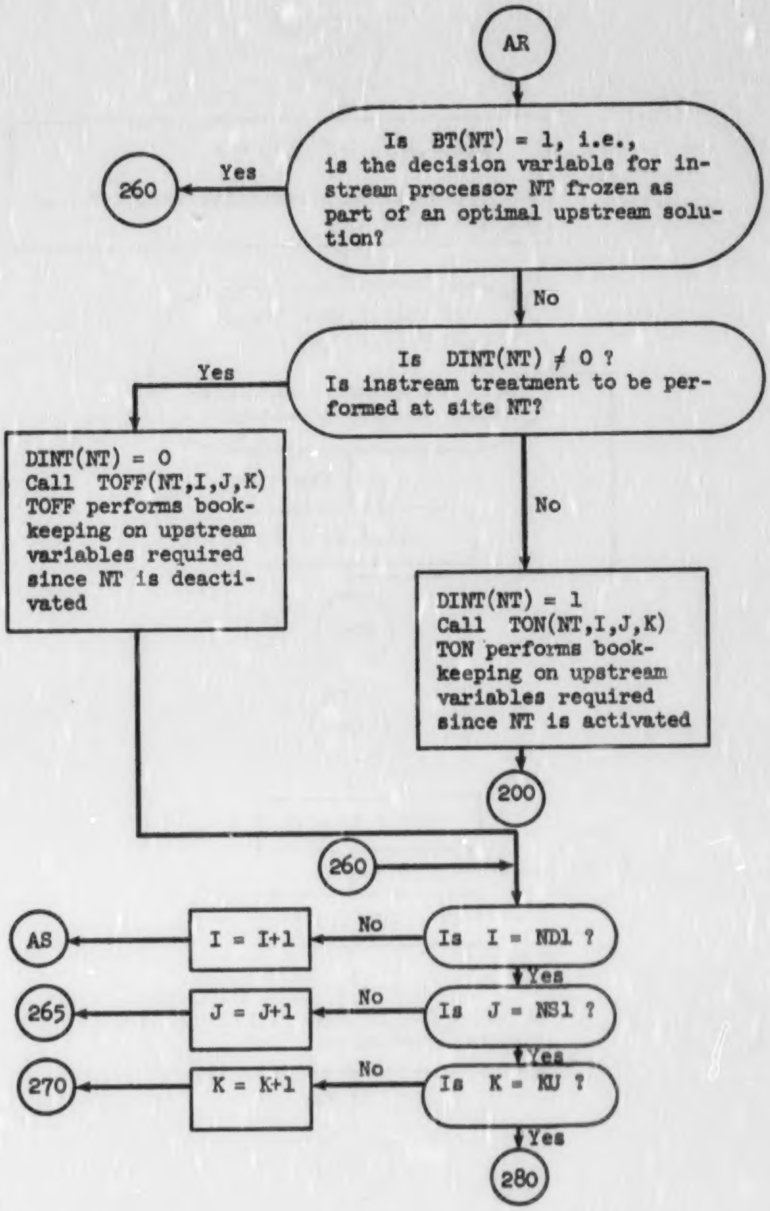
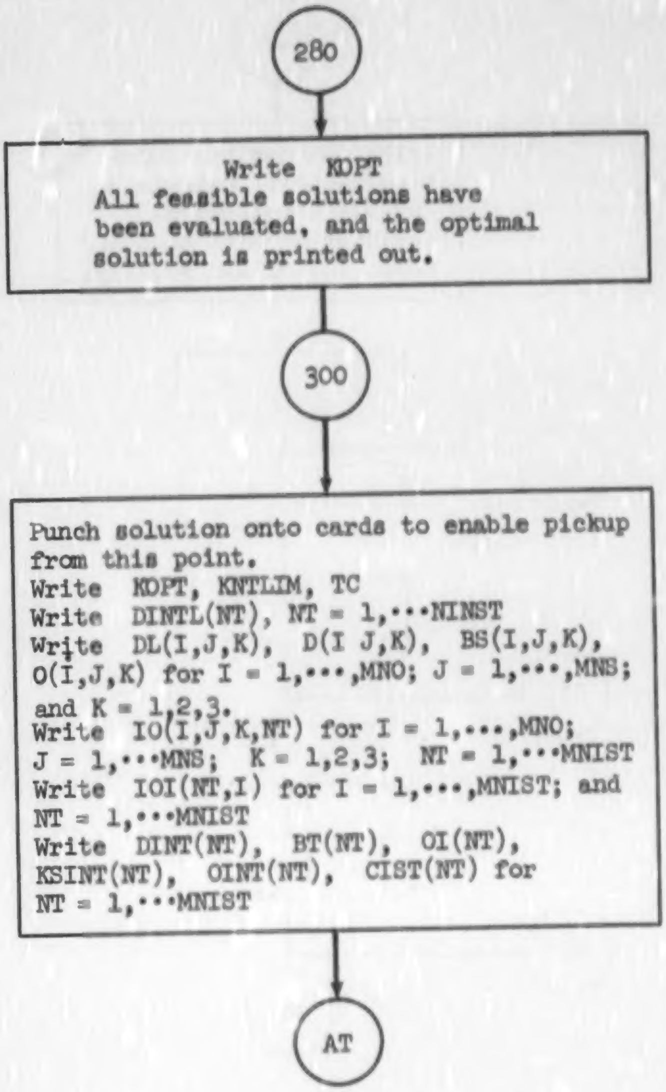

AR
Is BT(NT) = 1, i.e., is the decision variable for instream processor NT frozen as part of an optimal upstream solution?
Yes
260
No
Is DINT(NT) ≠ 0 ? Is instream treatment to be performed at site NT?
Yes
DINT(NT) = 0
Call TOFF(NT,I,J,K)
TOFF performs bookkeeping on upstream variables required since NT is deactivated
No
DINT(NT) = 1
Call TON(NT,I,J,K)
TON performs bookkeeping on upstream variables required since NT is activated
200
260
Is I = ND1 ?
No
I = I+1
AS
Yes
Is J = NS1 ?
No
J = J+1
265
Yes
Is K = KU ?
No
K = K+1
270
Yes
280

Figure C.14 Program ALCOT



Write KOPT
All feasible solutions have
been evaluated, and the optimal
solution is printed out.

Punch solution onto cards to enable pickup
from this point.
Write KOPT, KNTLIM, TC
Write DINTL(NT), NT = 1,•••NINST
Write DL(I,J,K), D(I J,K), BS(I,J,K),
O(I,J,K) for I = 1,•••,MNO; J = 1,•••,MNS;
and K = 1,2,3.
Write IO(I,J,K,NT) for I = 1,•••,MNO;
J = 1,•••MNS; K = 1,2,3; NT = 1,•••MNIST
Write IOI(NT,I) for I = 1,•••,MNIST; and
NT = 1,•••MNIST
Write DINT(NT), BT(NT), OI(NT),
KSINT(NT), OINT(NT), CIST(NT) for
NT = 1,•••MNIST

Figure C.14 Program ALCOT

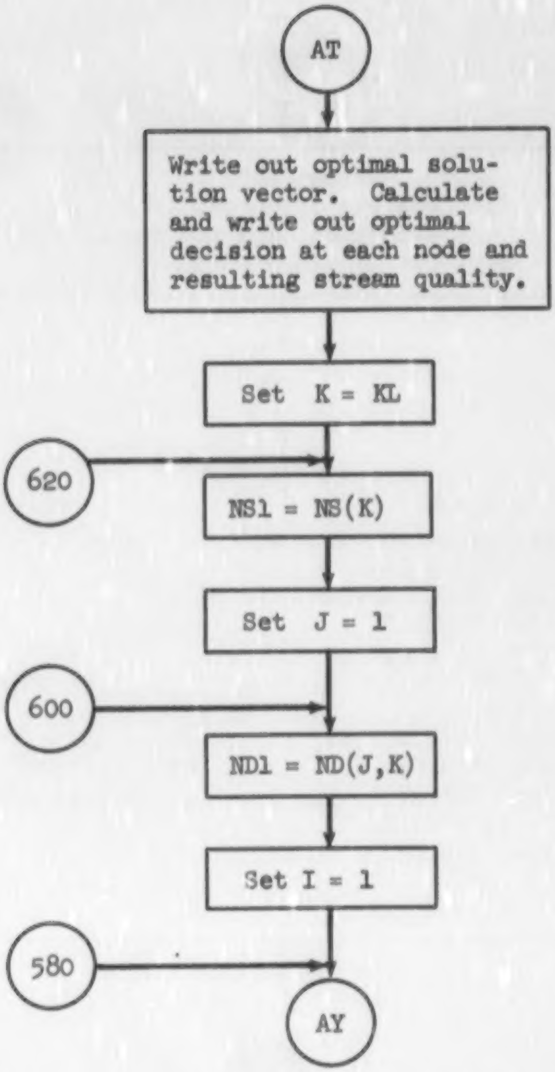

AT
Write out optimal solution vector. Calculate and write out optimal decision at each node and resulting stream quality.
Set K = KL
620
NS1 = NS(K)
Set J = 1
600
ND1 = ND(J,K)
Set I = 1
580
AY

Figure C.14 Program ALCOT

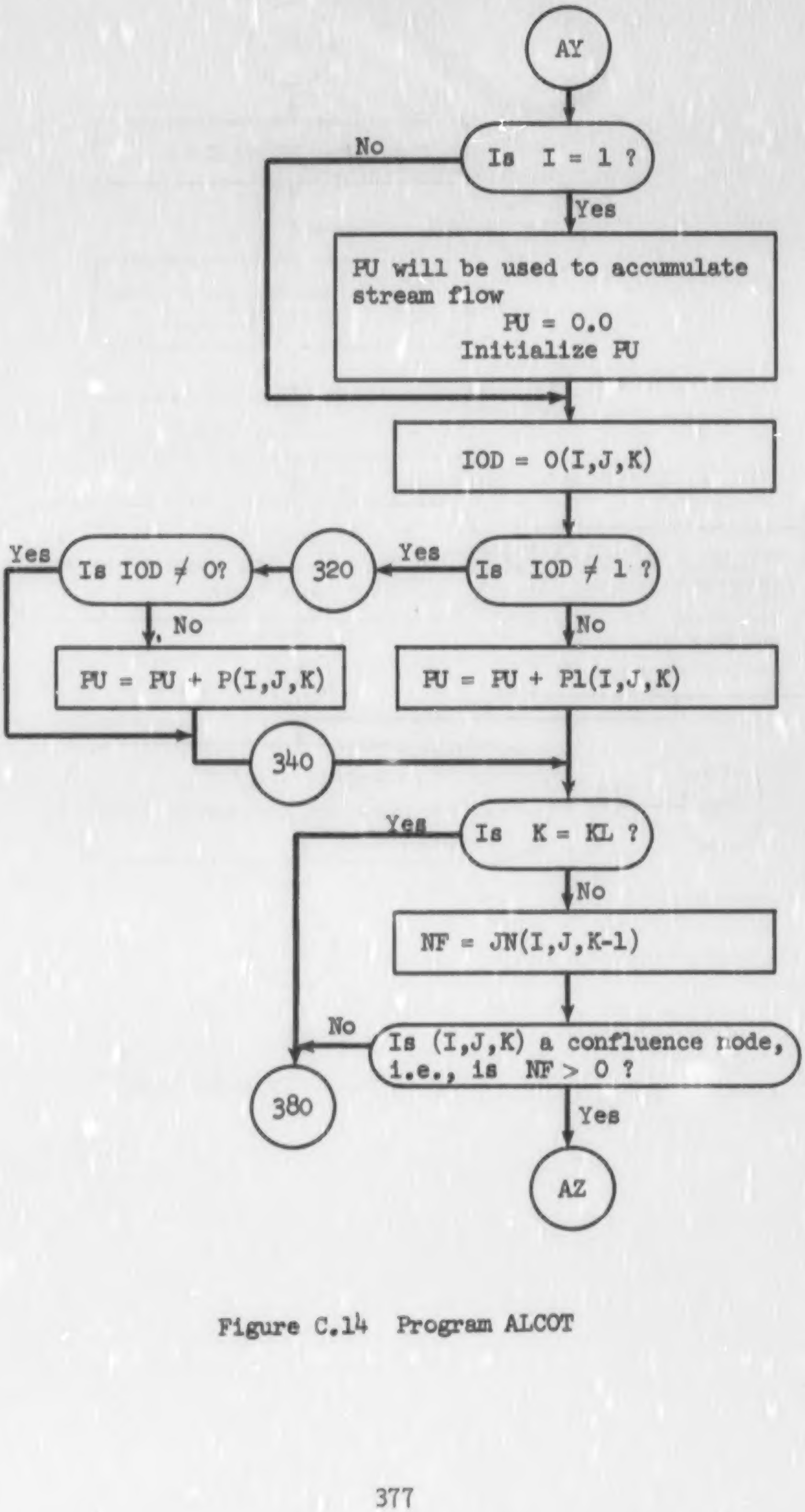

AY
Is I = 1 ?
No
Yes
PU will be used to accumulate stream flow
PU = 0.0
Initialize PU
IOD = O(I,J,K)
Yes
Is IOD ≠ 0?
320
Yes
Is IOD ≠ 1 ?
No
No
PU = PU + P(I,J,K)
PU = PU + P1(I,J,K)
340
Yes
Is K = KL ?
No
NF = JN(I,J,K-1)
No
Is (I,J,K) a confluence node, i.e., is NF > 0 ?
380
Yes
AZ

Figure C.14 Program ALCOT

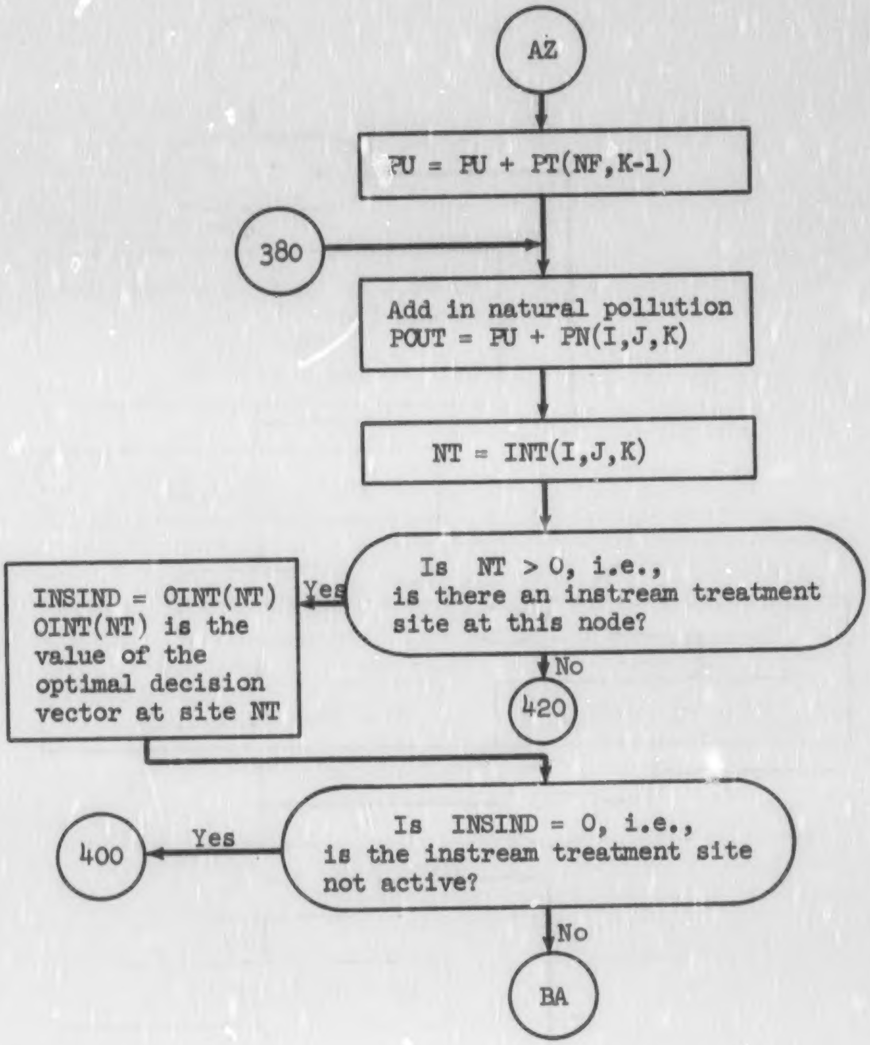

AZ
PU = PU + PT(NF,K-1)
380
Add in natural pollution
POUT = PU + PN(I,J,K)
NT = INT(I,J,K)
Is NT > 0, i.e.,
is there an instream treatment site at this node?
Yes
INSIND = OINT(NT)
OINT(NT) is the value of the optimal decision vector at site NT
No
420
Is INSIND = 0, i.e.,
is the instream treatment site not active?
Yes
400
No
BA

Figure C.14 Program ALCOT

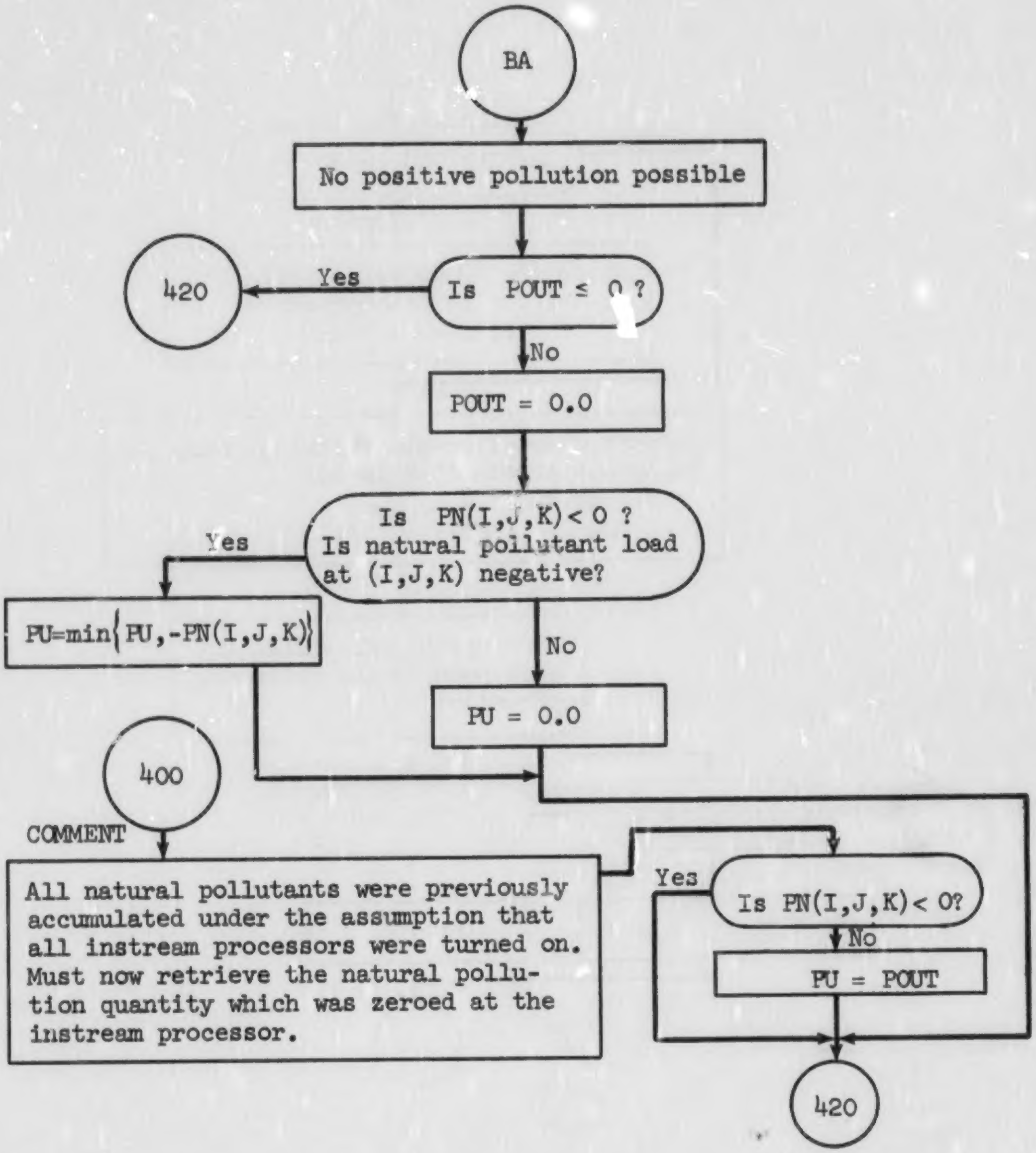

BA
No positive pollution possible
420
Yes
Is POUT ≤ 0 ?
No
POUT = 0.0
Is PN(I,J,K) < 0 ?
Is natural pollutant load at (I,J,K) negative?
Yes
PU=min{PU,-PN(I,J,K)}
No
PU = 0.0
400
COMMENT
All natural pollutants were previously accumulated under the assumption that all instream processors were turned on. Must now retrieve the natural pollution quantity which was zeroed at the instream processor.
Yes
Is PN(I,J,K) < 0?
No
PU = POUT
420

Figure C.14 Program ALCOT

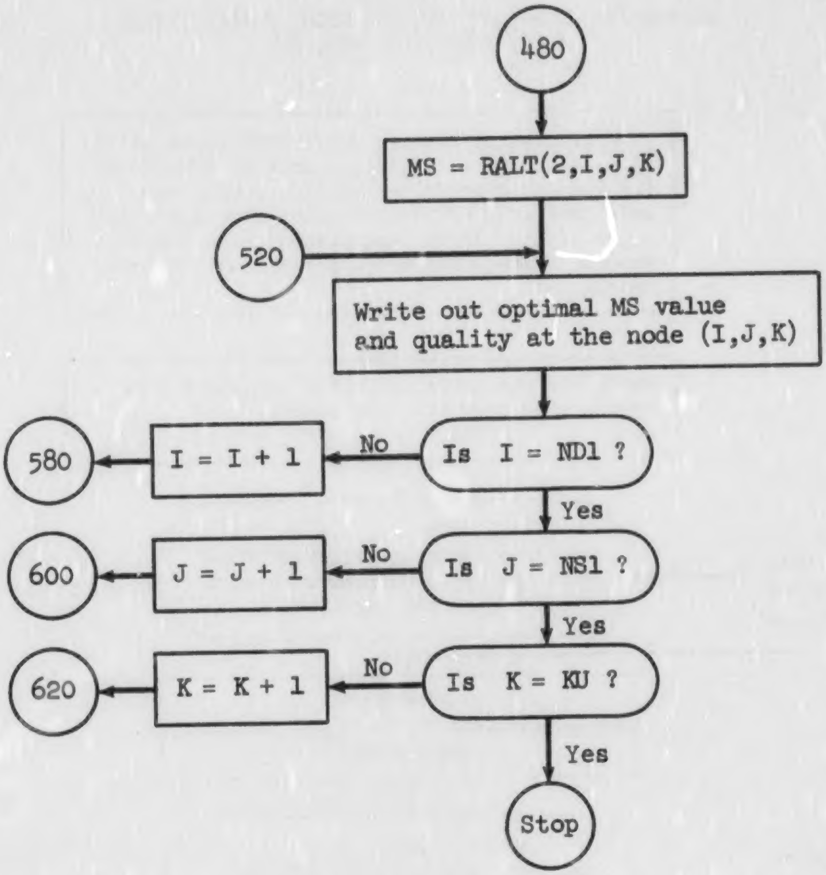
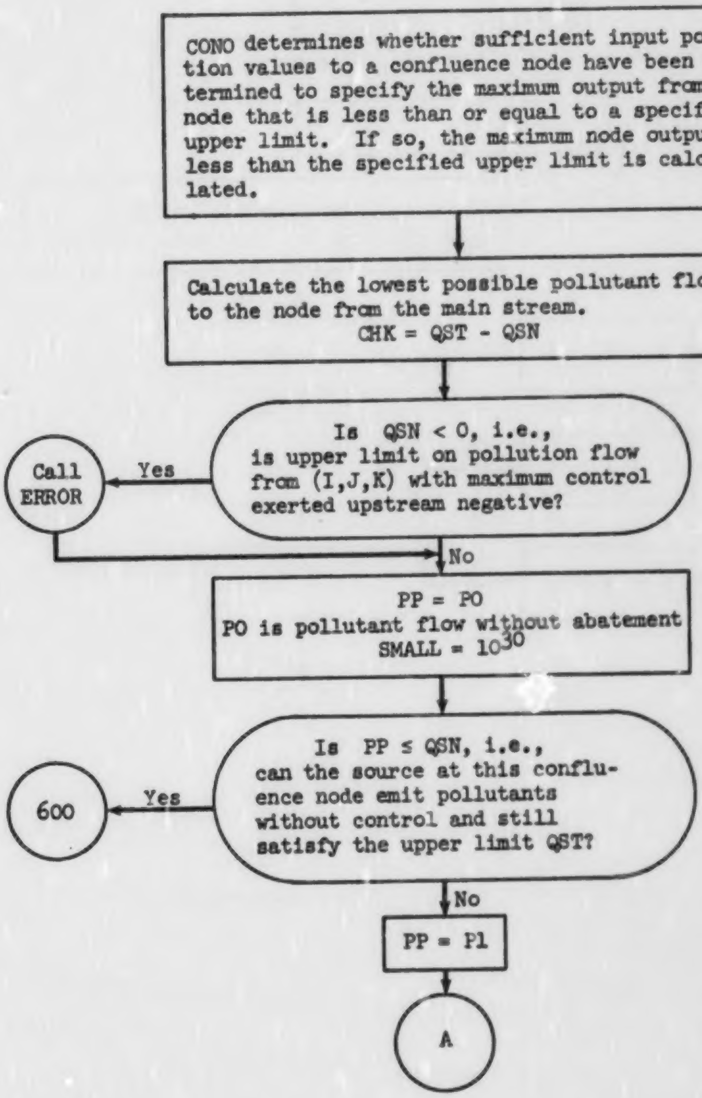

480
MS = RALT(2,I,J,K)
520
Write out optimal MS value and quality at the node (I,J,K)
Is I = ND1 ?
No
I = I + 1
580
Yes
Is J = NS1 ?
No
J = J + 1
600
Yes
Is K = KU ?
No
K = K + 1
620
Yes
Stop

Figure C.14 Program ALCOT

Subroutine CONO(QST, PO, P1, NDIS, PO1DIS, PJ1DIS, NREP, PLTM, PUS, QSN, CHO)



CONO determines whether sufficient input pollution values to a confluence node have been determined to specify the maximum output from the node that is less than or equal to a specified upper limit. If so, the maximum node output less than the specified upper limit is calculated.

Calculate the lowest possible pollutant flow to the node from the main stream.
$CHK = QST - QSN$

Is $QSN < 0$, i.e., is upper limit on pollution flow from (I,J,K) with maximum control exerted upstream negative?

$PP = PO$
PO is pollutant flow without abatement
$SMALL = 10^{30}$

Is $PP \le QSN$, i.e., can the source at this confluence node emit pollutants without control and still satisfy the upper limit QST?

$PP = P1$

Figure C.15 Subroutine CONO

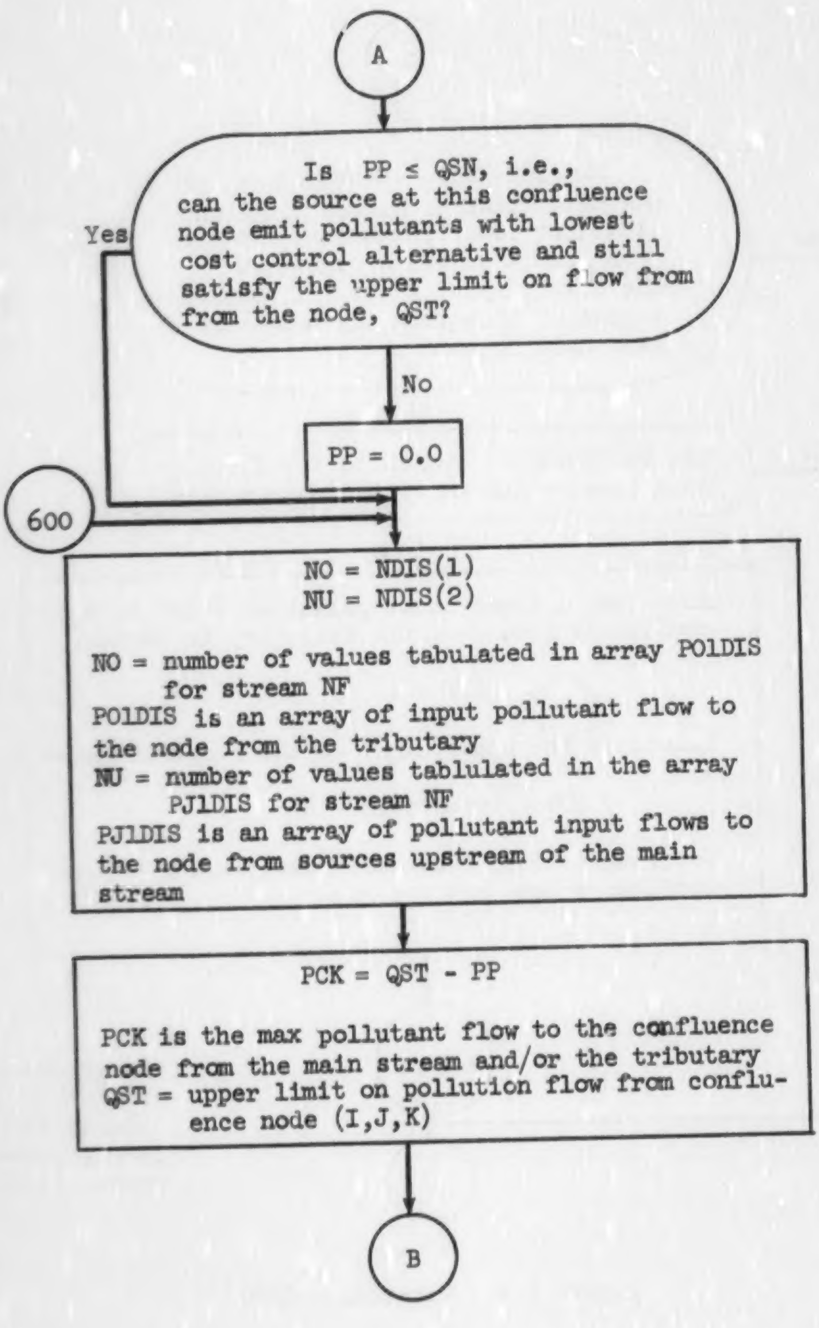
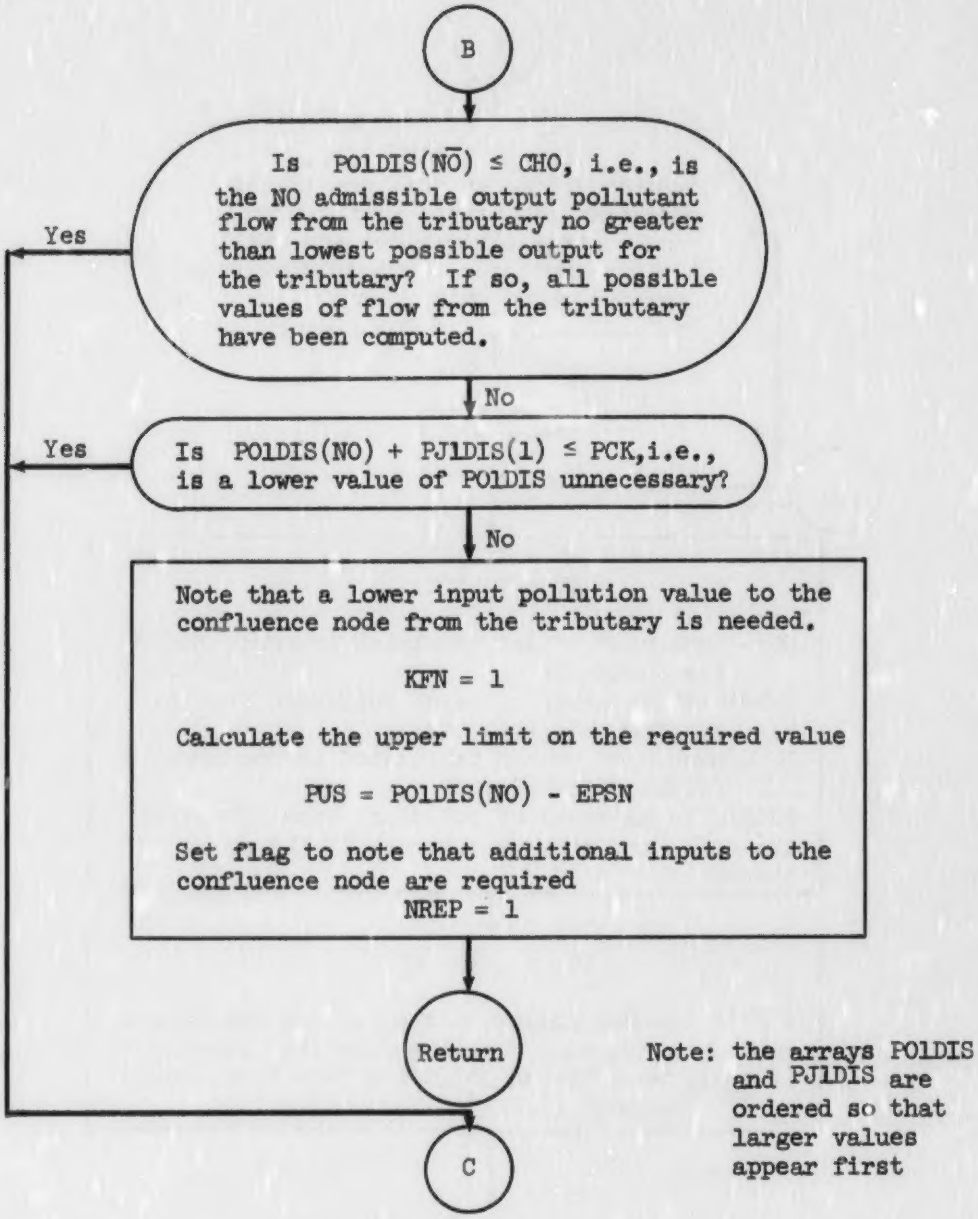

A
Is PP ≤ QSN, i.e., can the source at this confluence node emit pollutants with lowest cost control alternative and still satisfy the upper limit on flow from from the node, QST?
Yes
No
PP = 0.0
600
NO = NDIS(1)
NU = NDIS(2)
NO = number of values tabulated in array PO1DIS for stream NF
PO1DIS is an array of input pollutant flow to the node from the tributary
NU = number of values tablulated in the array PJ1DIS for stream NF
PJ1DIS is an array of pollutant input flows to the node from sources upstream of the main stream
PCK = QST - PP
PCK is the max pollutant flow to the confluence node from the main stream and/or the tributary
QST = upper limit on pollution flow from confluence node (I,J,K)
B

Figure C.15 Subroutine CONO


B
Is $\text{P01DIS}(\overline{\text{NO}}) \le \text{CHO}$, i.e., is the NO admissible output pollutant flow from the tributary no greater than lowest possible output for the tributary? If so, all possible values of flow from the tributary have been computed.
Yes
No
Is $\text{P01DIS(NO)} + \text{PJ1DIS}(1) \le \text{PCK}$, i.e., is a lower value of P01DIS unnecessary?
Yes
No
Note that a lower input pollution value to the confluence node from the tributary is needed.
KFN = 1
Calculate the upper limit on the required value
PUS = P01DIS(NO) - EPSN
Set flag to note that additional inputs to the confluence node are required
NREP = 1
Return
C

Note: the arrays P01DIS and PJ1DIS are ordered so that larger values appear first

Figure C.15 Subroutine CONO

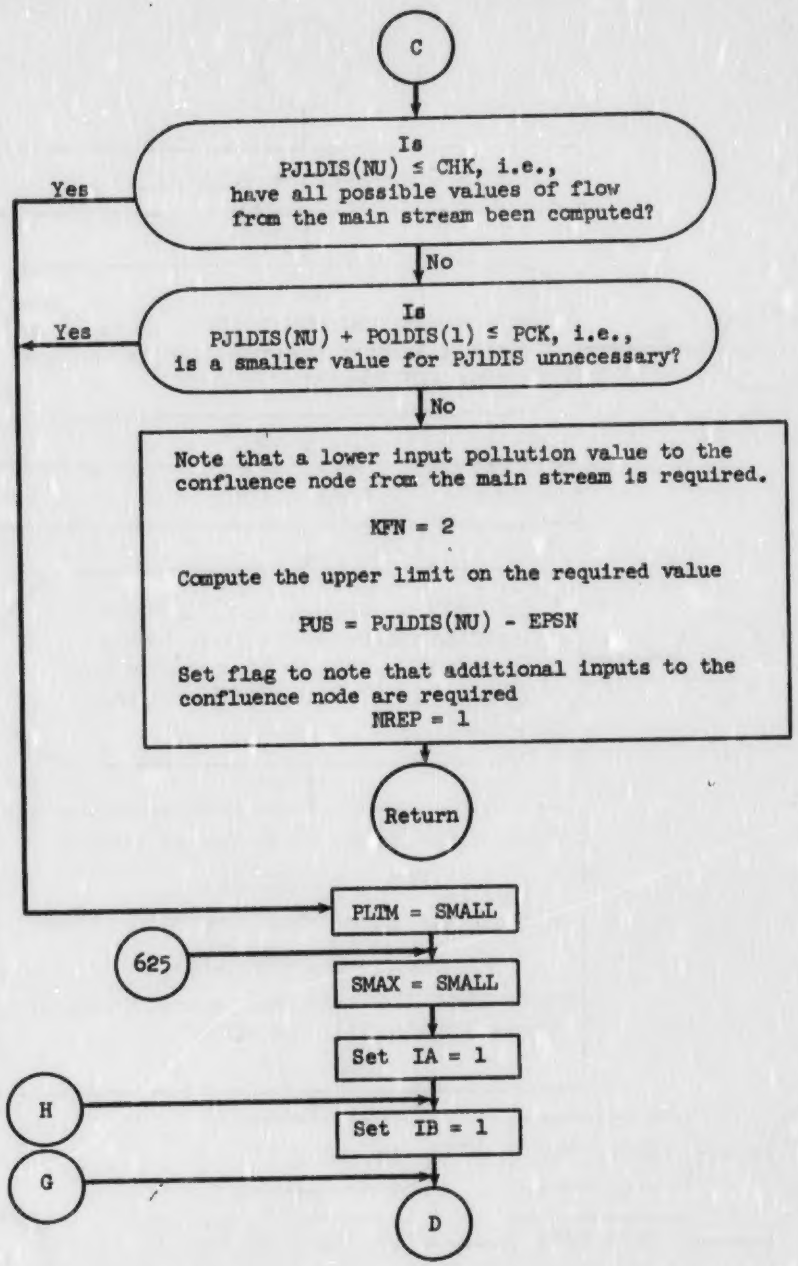

C
Is
PJ1DIS(NU) ≤ CHK, i.e.,
have all possible values of flow
from the main stream been computed?
Yes
No
Is
PJ1DIS(NU) + PO1DIS(1) ≤ PCK, i.e.,
is a smaller value for PJ1DIS unnecessary?
Yes
No
Note that a lower input pollution value to the confluence node from the main stream is required.
KFN = 2
Compute the upper limit on the required value
PUS = PJ1DIS(NU) - EPSN
Set flag to note that additional inputs to the confluence node are required
NREP = 1
Return
PLTM = SMALL
625
SMAX = SMALL
Set IA = 1
H
Set IB = 1
G
D

Figure C.15 Subroutine CONO

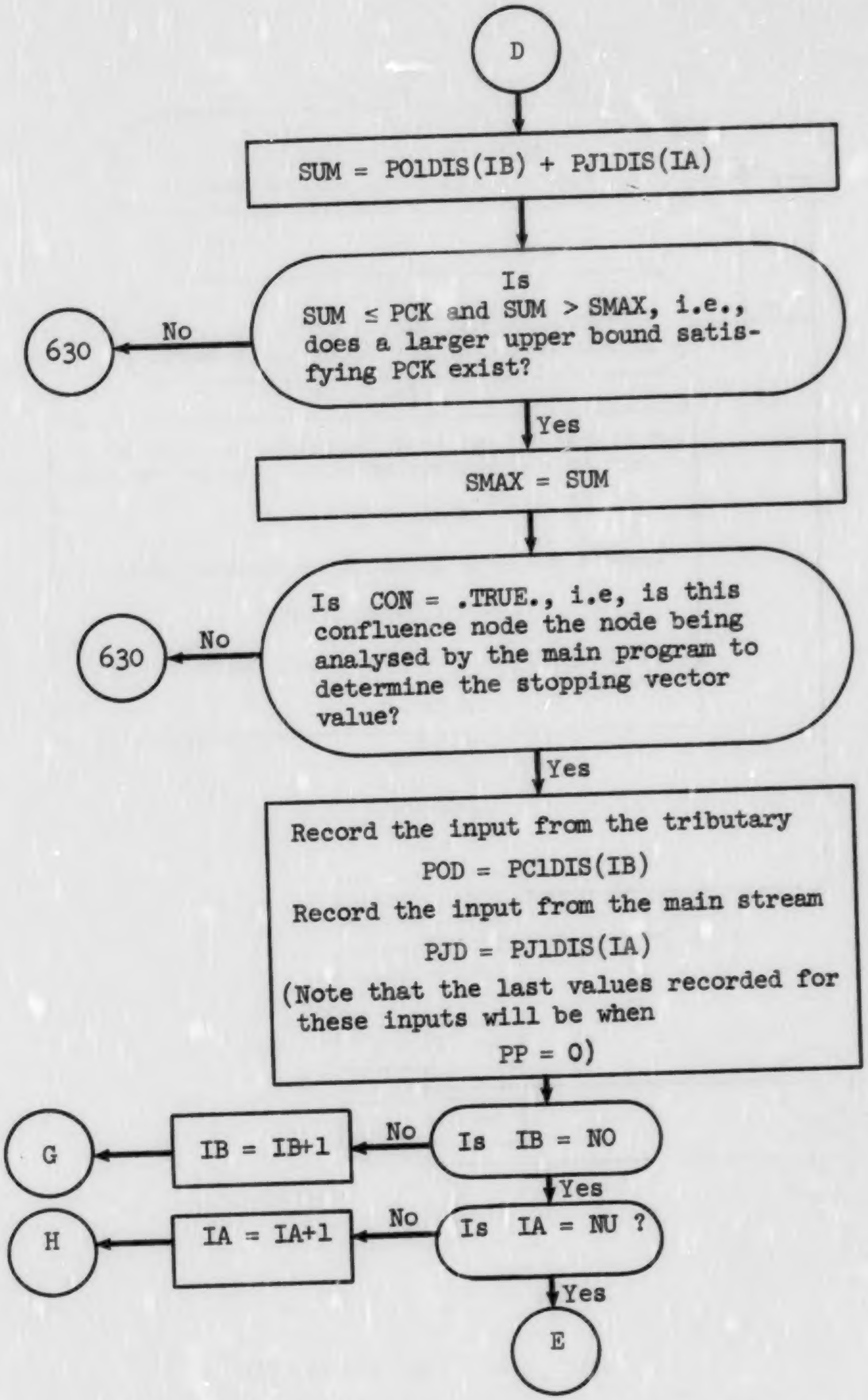

D
SUM = PO1DIS(IB) + PJ1DIS(IA)
Is
SUM ≤ PCK and SUM > SMAX, i.e., does a larger upper bound satisfying PCK exist?
No
630
Yes
SMAX = SUM
Is CON = .TRUE., i.e, is this confluence node the node being analysed by the main program to determine the stopping vector value?
No
630
Yes
Record the input from the tributary
POD = PC1DIS(IB)
Record the input from the main stream
PJD = PJ1DIS(IA)
(Note that the last values recorded for these inputs will be when
PP = 0)
Is IB = NO
No
IB = IB+1
G
Yes
Is IA = NU ?
No
IA = IA+1
H
Yes
E

Figure C.15 Subroutine CONO

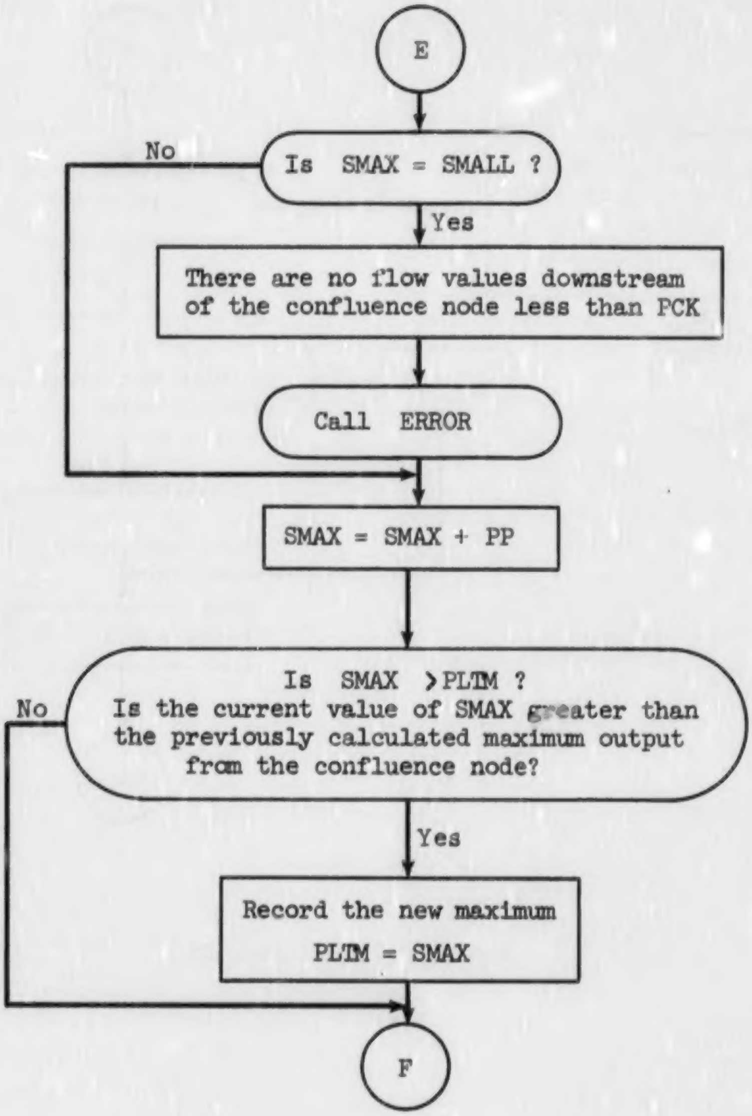
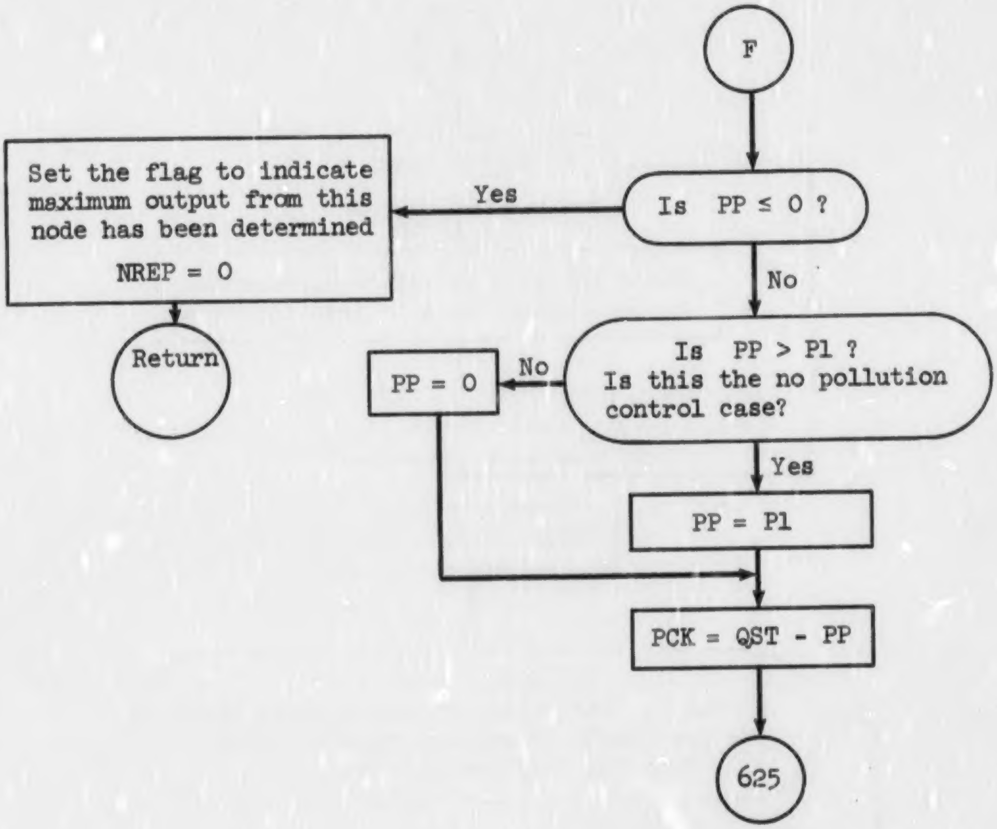

E
Is SMAX = SMALL ?
No
Yes
There are no flow values downstream of the confluence node less than PCK
Call ERROR
SMAX = SMAX + PP
Is SMAX >PLTM ?
Is the current value of SMAX greater than the previously calculated maximum output from the confluence node?
No
Yes
Record the new maximum
PLTM = SMAX
F

Figure C.15 Subroutine CONO



F
Is PP ≤ 0 ?
Yes
Set the flag to indicate maximum output from this node has been determined
NREP = 0
Return
No
Is PP > P1 ?
Is this the no pollution control case?
No
PP = 0
Yes
PP = P1
PCK = QST - PP
625

Figure C.15 Subroutine

Subroutine ERROR

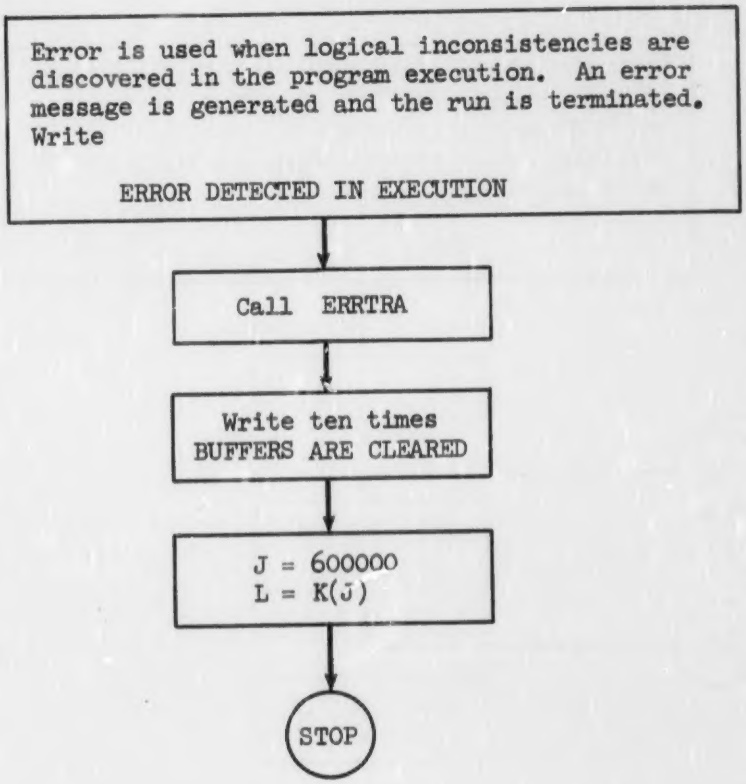
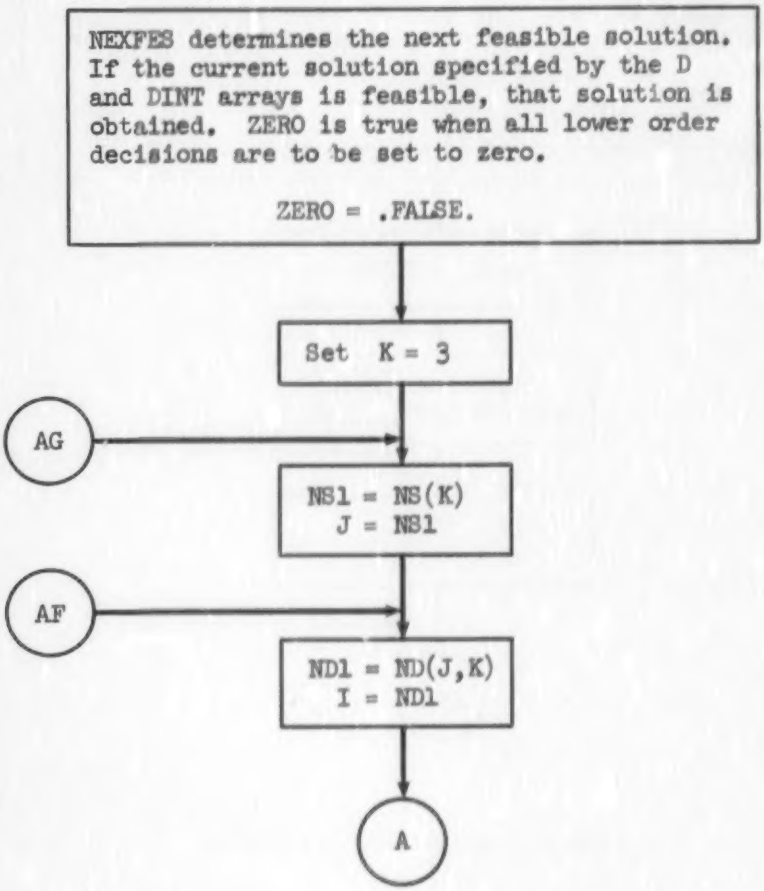

Error is used when logical inconsistencies are discovered in the program execution. An error message is generated and the run is terminated. Write
ERROR DETECTED IN EXECUTION
Call ERRTRA
Write ten times
BUFFERS ARE CLEARED
J = 600000
L = K(J)
STOP

Figure C.16 Subroutine ERROR

Subroutine NEXFES



NEXFES determines the next feasible solution. If the current solution specified by the D and DINT arrays is feasible, that solution is obtained. ZERO is true when all lower order decisions are to be set to zero.

ZERO = .FALSE.
Set K = 3
AG
NS1 = NS(K)
J = NS1
AF
ND1 = ND(J,K)
I = ND1
A

Figure C.17 Subroutine NEXFES

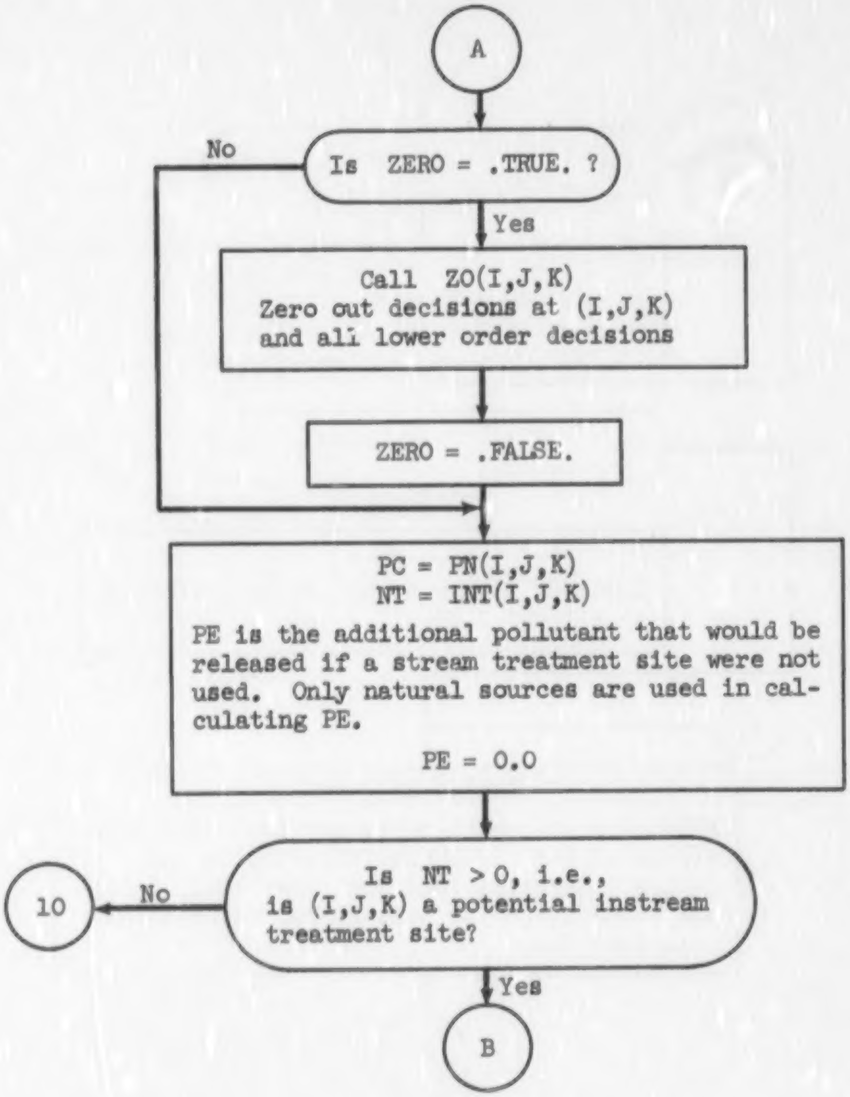

A
Is ZERO = .TRUE. ?
No
Yes
Call ZO(I,J,K)
Zero out decisions at (I,J,K)
and all lower order decisions
ZERO = .FALSE.
PC = PN(I,J,K)
NT = INT(I,J,K)
PE is the additional pollutant that would be released if a stream treatment site were not used. Only natural sources are used in calculating PE.
PE = 0.0
Is NT > 0, i.e.,
is (I,J,K) a potential instream treatment site?
No
10
Yes
B

Figure C.17 Subroutine NEXFES

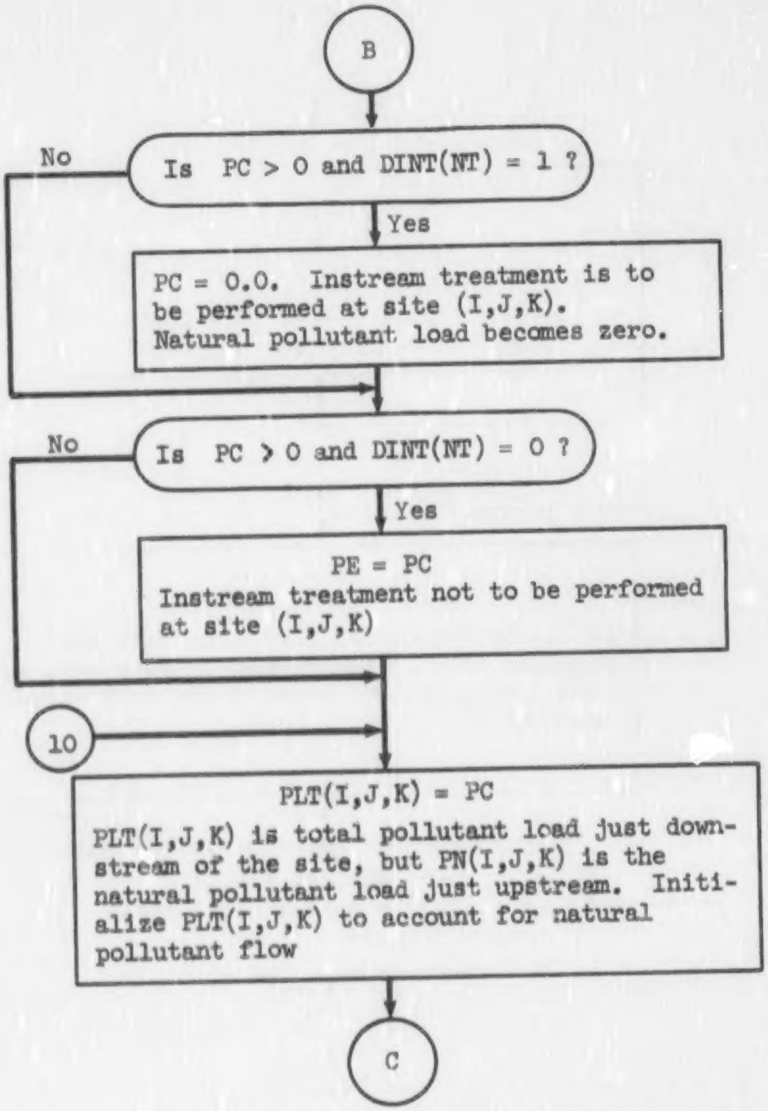

B
Is PC > 0 and DINT(NT) = 1 ?
No
Yes
PC = 0.0. Instream treatment is to be performed at site (I,J,K). Natural pollutant load becomes zero.
Is PC > 0 and DINT(NT) = 0 ?
No
Yes
PE = PC
Instream treatment not to be performed at site (I,J,K)
10
PLT(I,J,K) = PC
PLT(I,J,K) is total pollutant load just downstream of the site, but PN(I,J,K) is the natural pollutant load just upstream. Initialize PLT(I,J,K) to account for natural pollutant flow
C

Figure C.17 Subroutine NEXFES

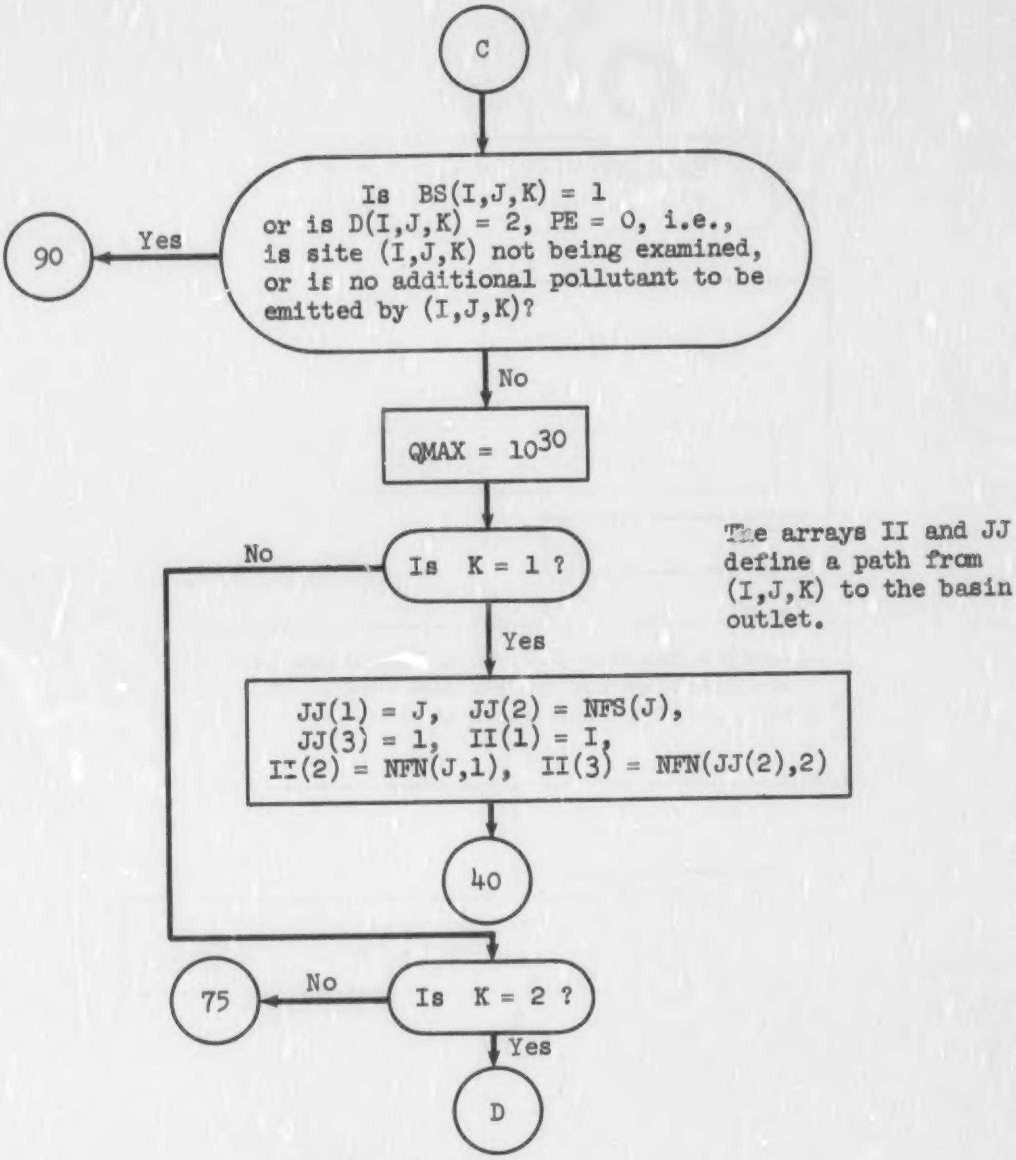

C
Is BS(I,J,K) = 1
or is D(I,J,K) = 2, PE = 0, i.e.,
is site (I,J,K) not being examined,
or is no additional pollutant to be
emitted by (I,J,K)?
Yes
90
No
QMAX = 10^30
Is K = 1 ?
No
Yes
The arrays II and JJ define a path from (I,J,K) to the basin outlet.
JJ(1) = J, JJ(2) = NFS(J),
JJ(3) = 1, II(1) = I,
II(2) = NFN(J,1), II(3) = NFN(JJ(2),2)
40
Is K = 2 ?
No
75
Yes
D

Figure C.17 Subroutine NEXFES

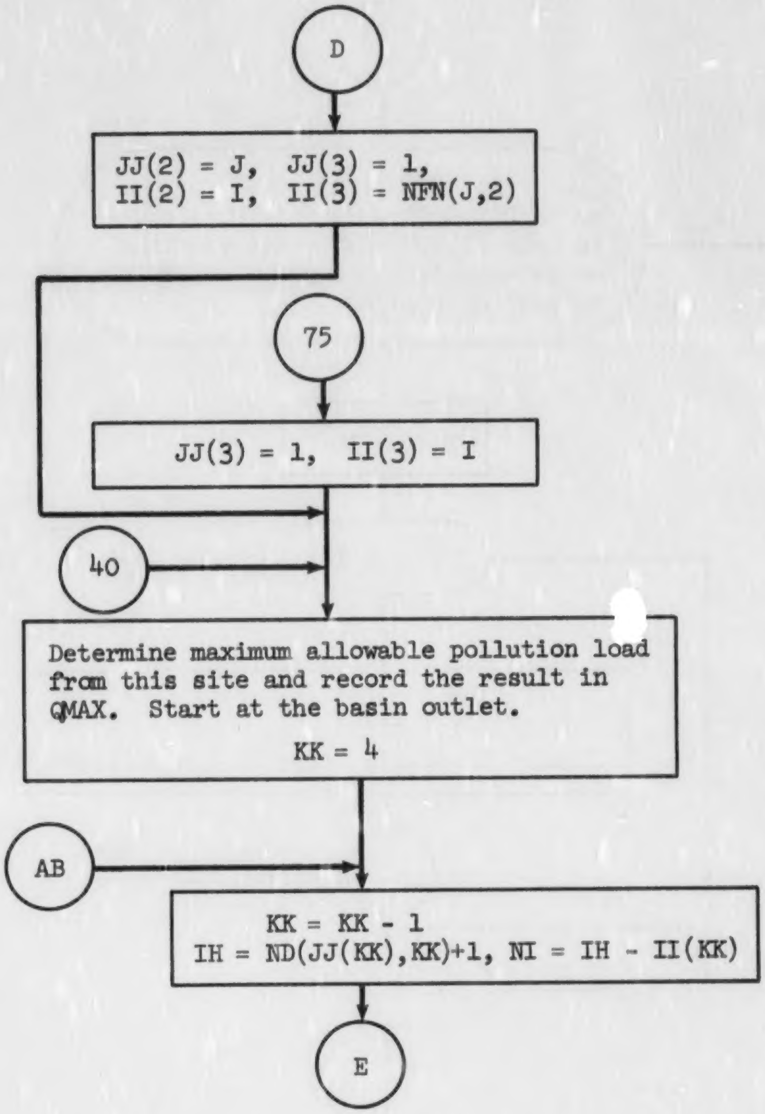

D
JJ(2) = J, JJ(3) = 1,
II(2) = I, II(3) = NFN(J,2)
75
JJ(3) = 1, II(3) = I
40
Determine maximum allowable pollution load from this site and record the result in QMAX. Start at the basin outlet.
KK = 4
AB
KK = KK - 1
IH = ND(JJ(KK),KK)+1, NI = IH - II(KK)
E

Figure C.17 Subroutine NEXFES

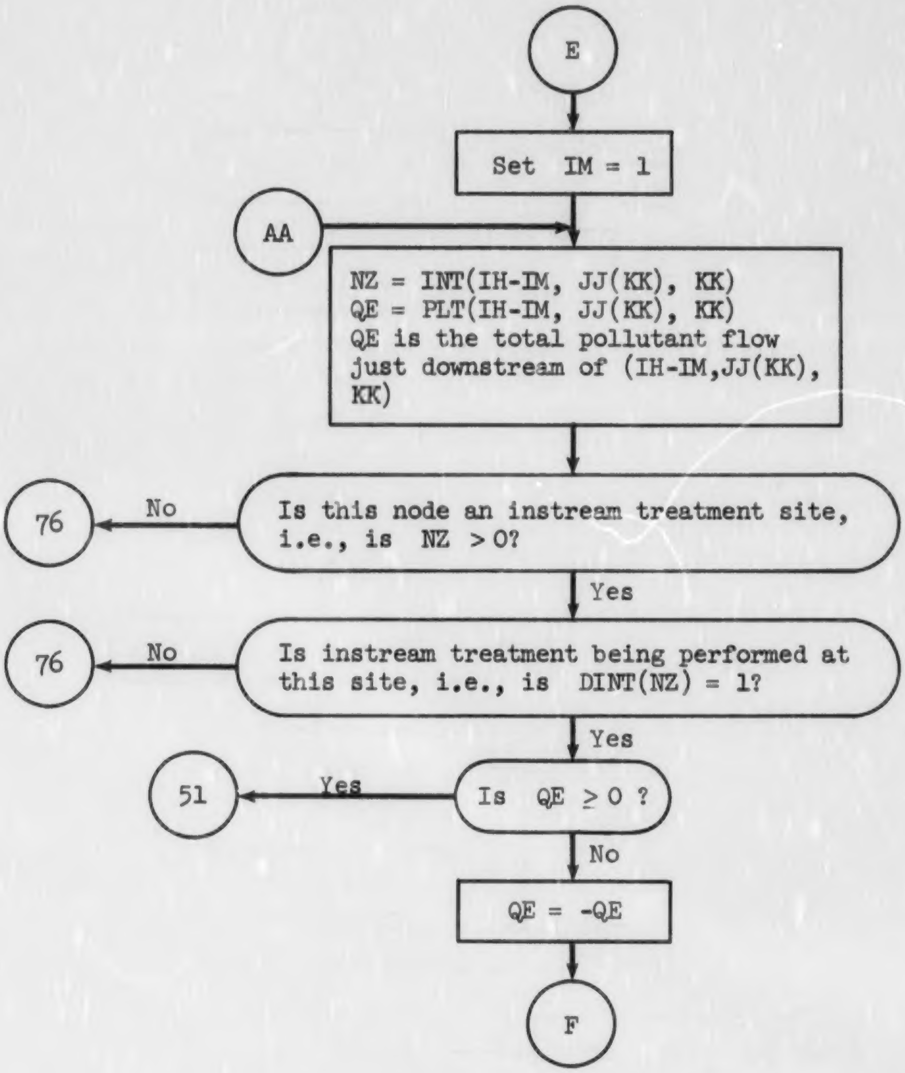

E
Set IM = 1
AA
NZ = INT(IH-IM, JJ(KK), KK)
QE = PLT(IH-IM, JJ(KK), KK)
QE is the total pollutant flow just downstream of (IH-IM,JJ(KK), KK)
76
No
Is this node an instream treatment site, i.e., is NZ > 0?
Yes
76
No
Is instream treatment being performed at this site, i.e., is DINT(NZ) = 1?
Yes
51
Yes
Is QE ≥ 0 ?
No
QE = -QE
F

Figure C.17 Subroutine NEXFES

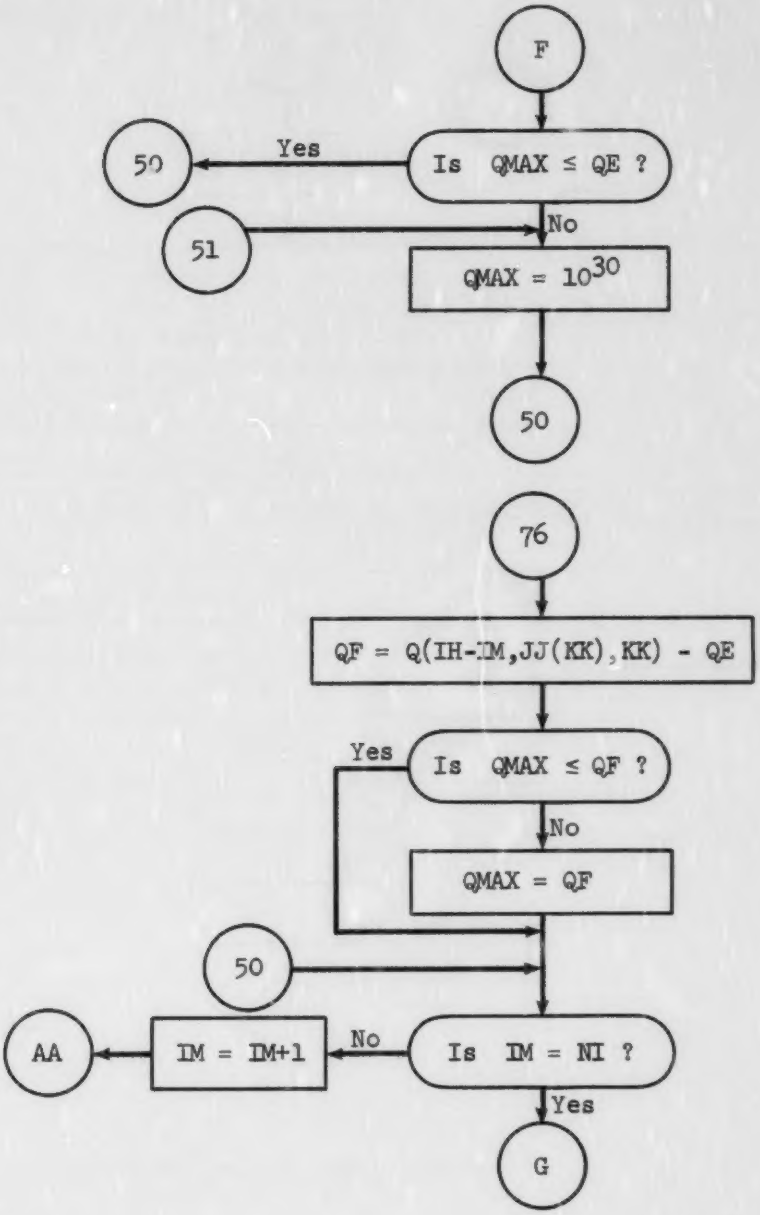

F
Is QMAX ≤ QE ?
Yes
50
No
51
QMAX = 10^30
50
76
QF = Q(IH-IM,JJ(KK),KK) - QE
Is QMAX ≤ QF ?
Yes
No
QMAX = QF
50
Is IM = NI ?
No
IM = IM+1
AA
Yes
G

Figure C.17 Subroutine NEXFES

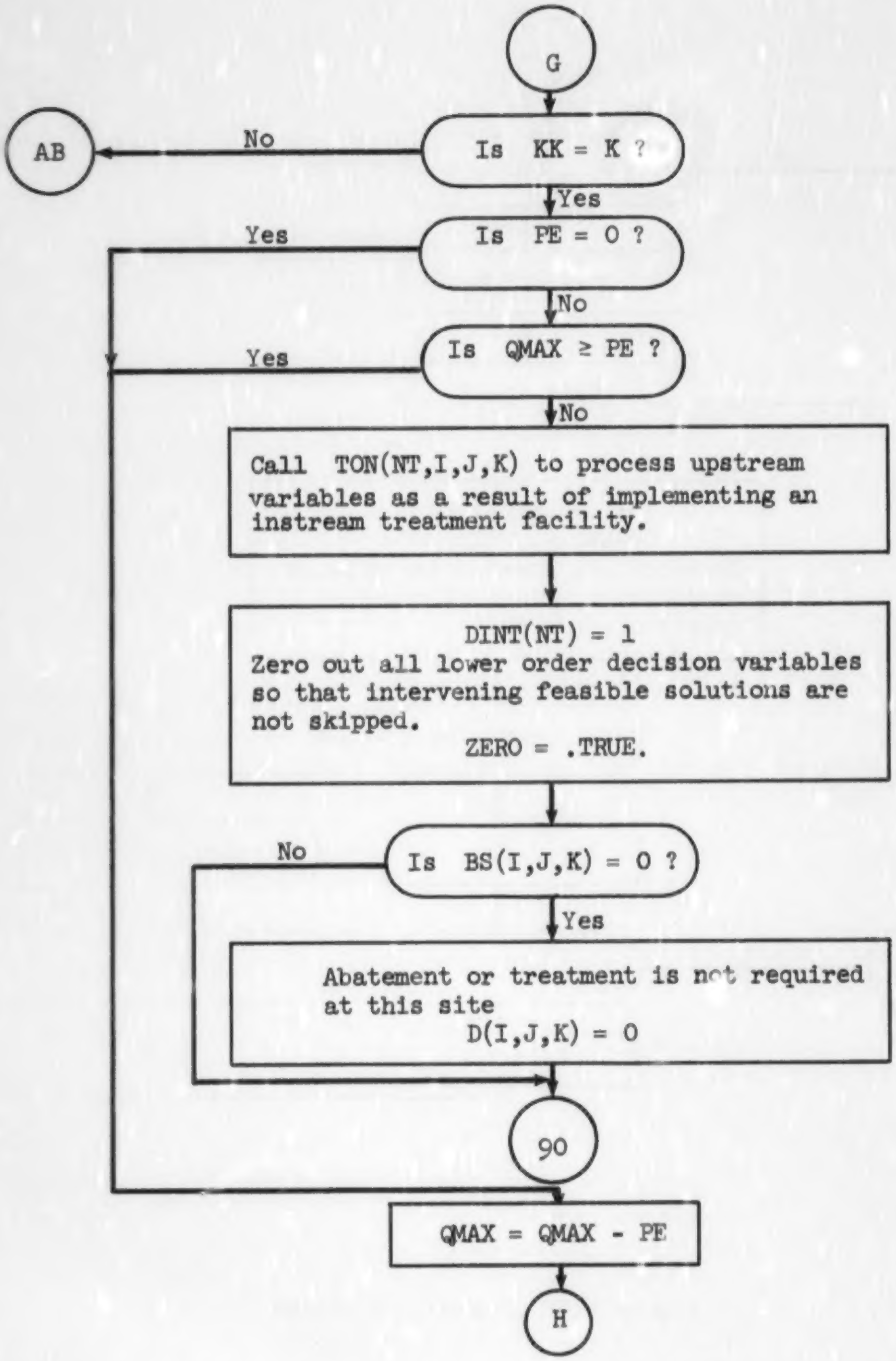

G
Is KK = K ?
No
AB
Yes
Is PE = 0 ?
Yes
No
Is QMAX ≥ PE ?
Yes
No
Call TON(NT,I,J,K) to process upstream variables as a result of implementing an instream treatment facility.
DINT(NT) = 1
Zero out all lower order decision variables so that intervening feasible solutions are not skipped.
ZERO = .TRUE.
Is BS(I,J,K) = 0 ?
No
Yes
Abatement or treatment is not required at this site
D(I,J,K) = 0
90
QMAX = QMAX - PE
H

Figure C.17 Subroutine NEXFES

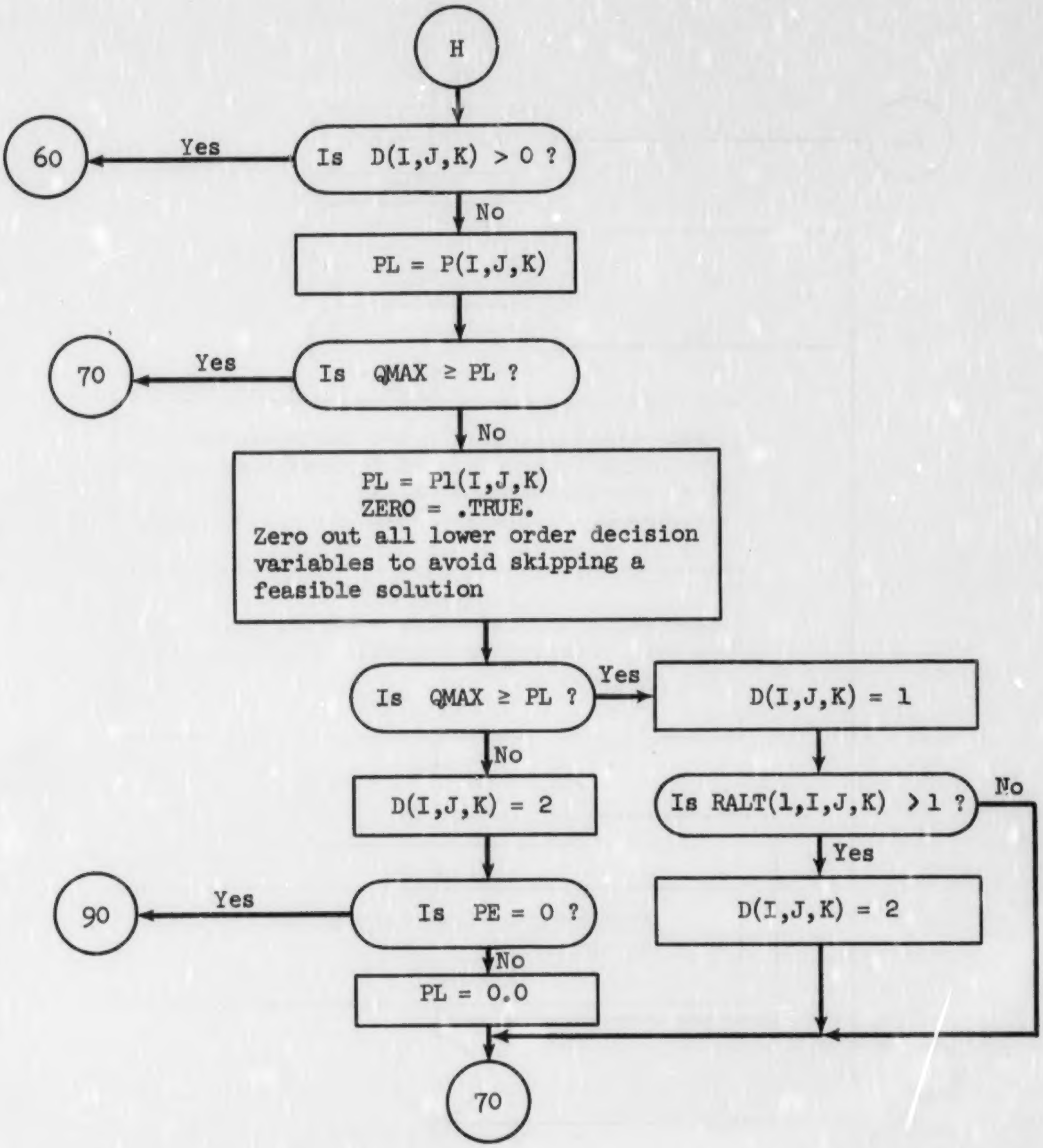

H
Is D(I,J,K) > 0 ?
Yes
60
No
PL = P(I,J,K)
Is QMAX ≥ PL ?
Yes
70
No
PL = Pl(I,J,K)
ZERO = .TRUE.
Zero out all lower order decision variables to avoid skipping a feasible solution
Is QMAX ≥ PL ?
Yes
D(I,J,K) = 1
No
D(I,J,K) = 2
Is RALT(1,I,J,K) > 1 ?
No
Yes
D(I,J,K) = 2
Is PE = 0 ?
Yes
90
No
PL = 0.0
70

Figure C.17 Subroutine NEXFES

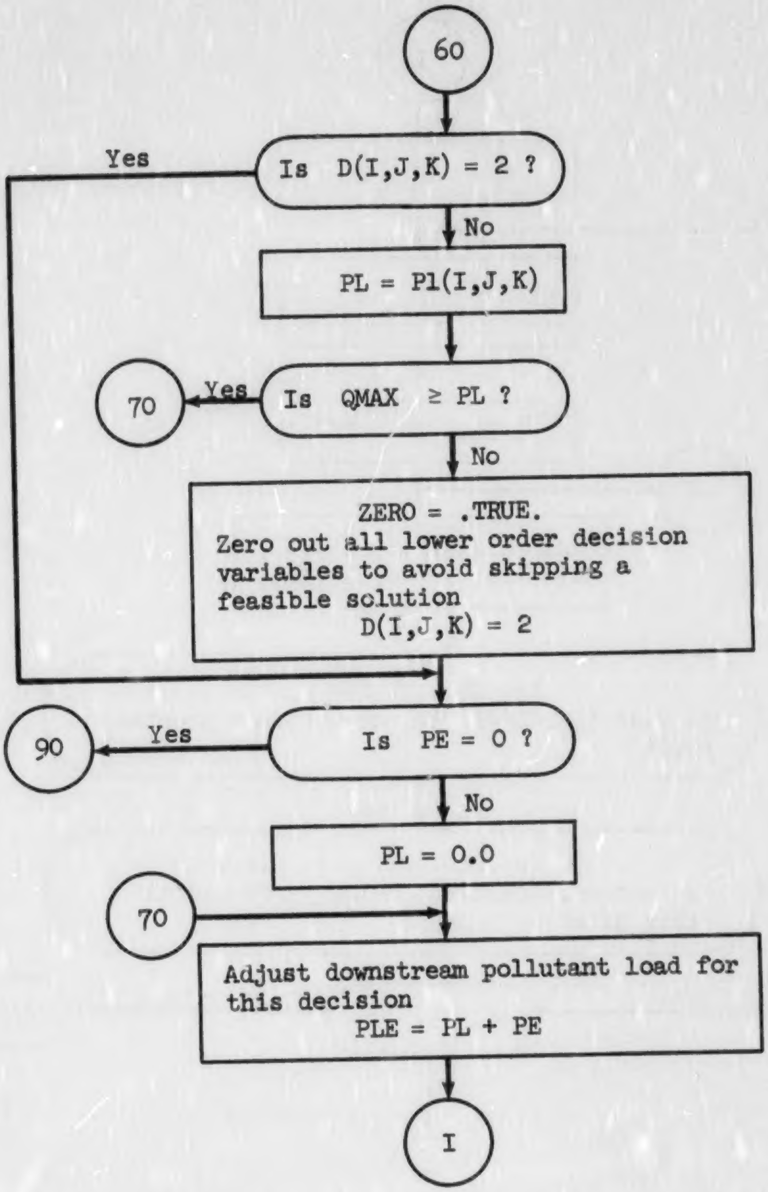

60
Is D(I,J,K) = 2 ?
Yes
No
PL = Pl(I,J,K)
Is QMAX ≥ PL ?
Yes
70
No
ZERO = .TRUE.
Zero out all lower order decision variables to avoid skipping a feasible solution
D(I,J,K) = 2
Is PE = 0 ?
Yes
90
No
PL = 0.0
70
Adjust downstream pollutant load for this decision
PLE = PL + PE
I

Figure C.17 Subroutine NEXFES

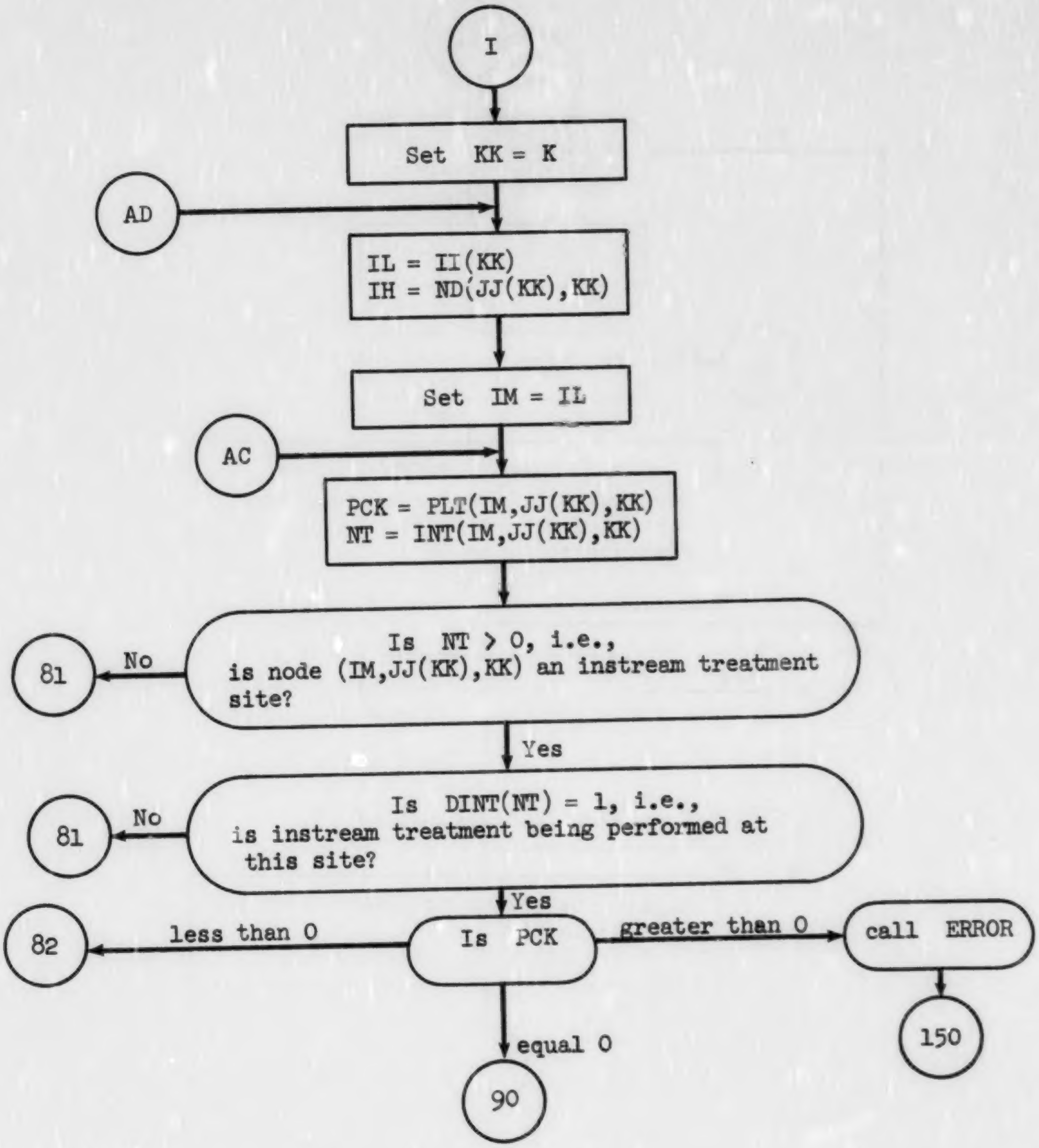

I
Set KK = K
AD
IL = II(KK)
IH = ND(JJ(KK),KK)
Set IM = IL
AC
PCK = PLT(IM,JJ(KK),KK)
NT = INT(IM,JJ(KK),KK)
Is NT > 0, i.e., is node (IM,JJ(KK),KK) an instream treatment site?
No
81
Yes
Is DINT(NT) = 1, i.e., is instream treatment being performed at this site?
No
81
Yes
Is PCK
less than 0
82
greater than 0
call ERROR
150
equal 0
90

Figure C.17 Subroutine NEXFES

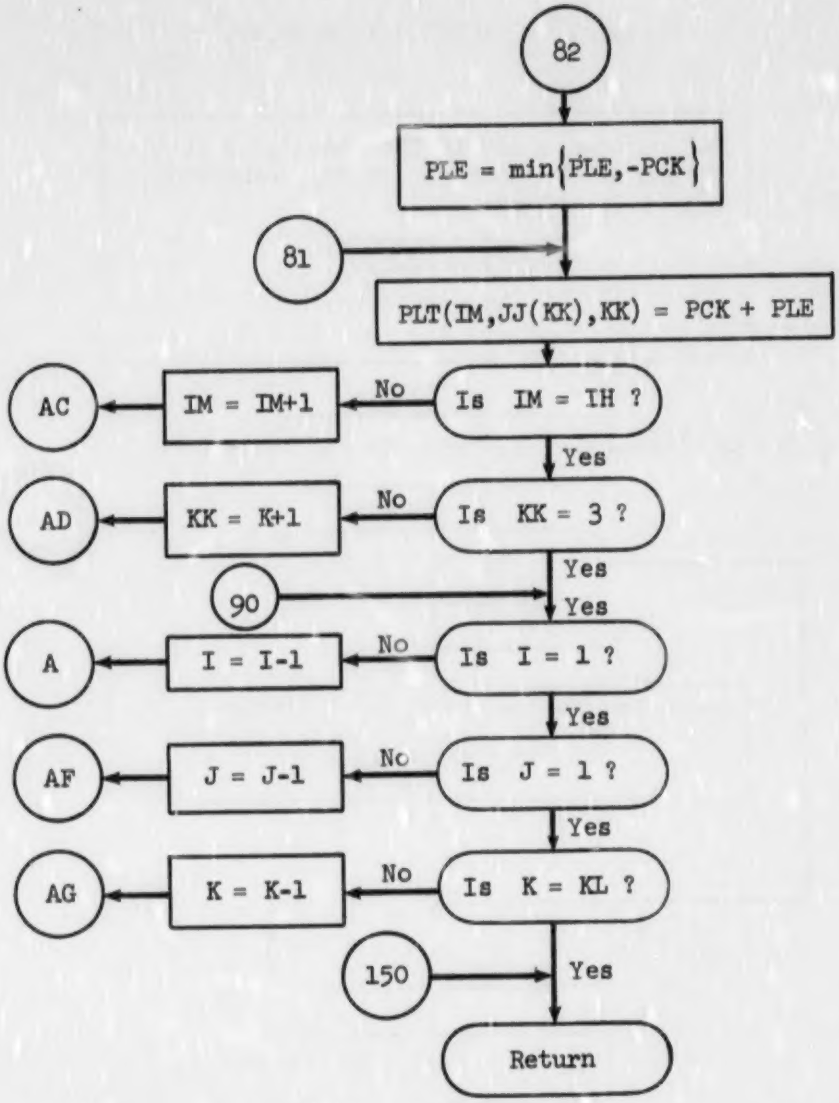

82
PLE = min{PLE,-PCK}
81
PLT(IM,JJ(KK),KK) = PCK + PLE
Is IM = IH ?
No
IM = IM+1
AC
Yes
Is KK = 3 ?
No
KK = K+1
AD
Yes
90
Yes
Is I = 1 ?
No
I = I-1
A
Yes
Is J = 1 ?
No
J = J-1
AF
Yes
Is K = KL ?
No
K = K-1
AG
150
Yes
Return

Figure C.17 Subroutine NEXFES

Function NON(I,J,K,IM,JM,KM)

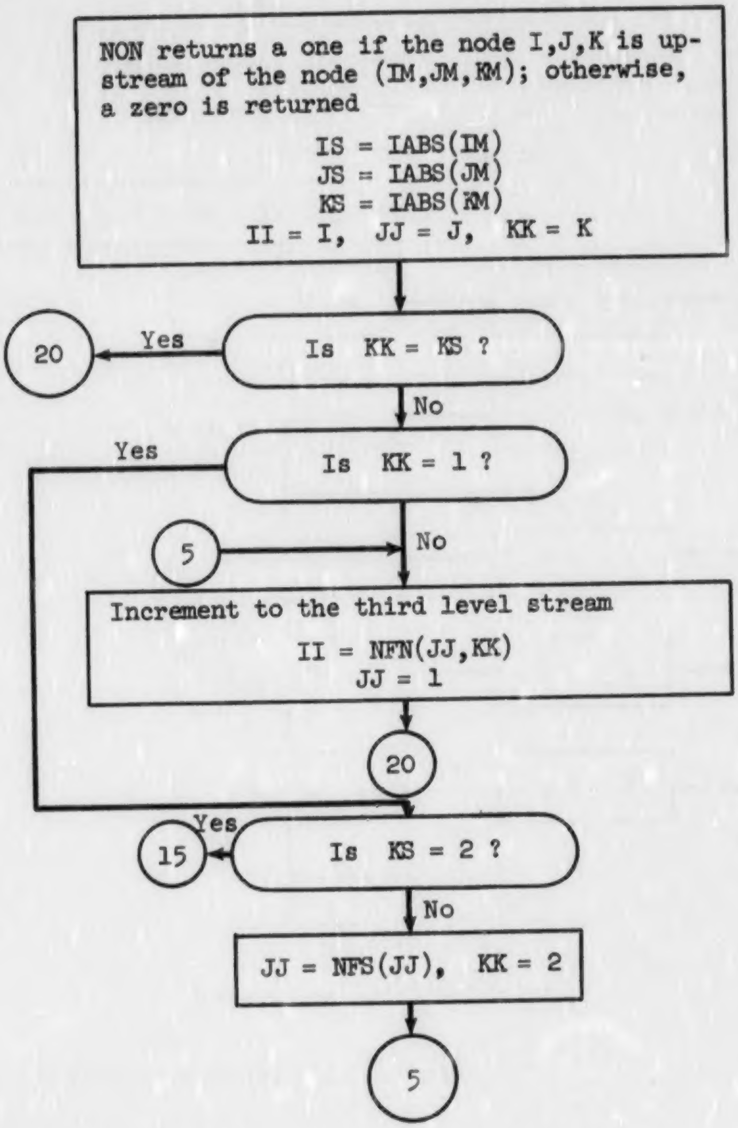

NON returns a one if the node I,J,K is upstream of the node (IM,JM,KM); otherwise, a zero is returned
IS = IABS(IM)
JS = IABS(JM)
KS = IABS(KM)
II = I, JJ = J, KK = K
Is KK = KS ?
Yes
20
No
Is KK = 1 ?
Yes
5
No
Increment to the third level stream
II = NFN(JJ,KK)
JJ = 1
20
Is KS = 2 ?
Yes
15
No
JJ = NFS(JJ), KK = 2
5

Figure C.18 Function NON

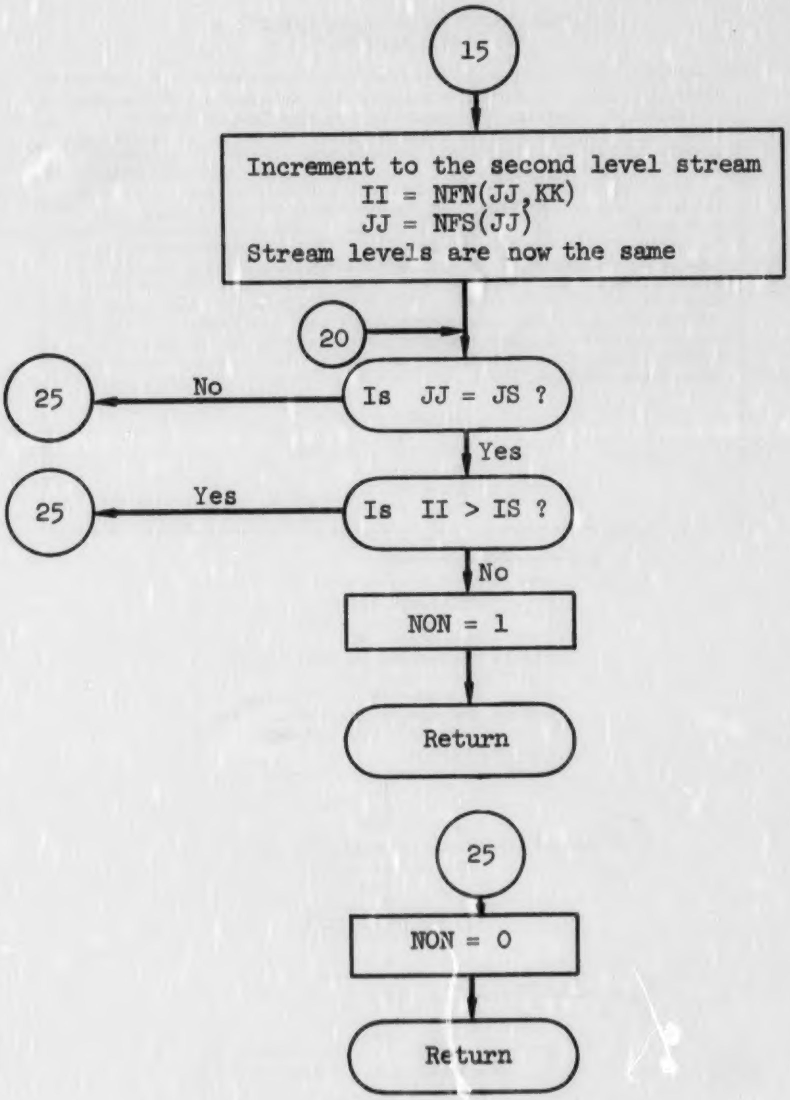
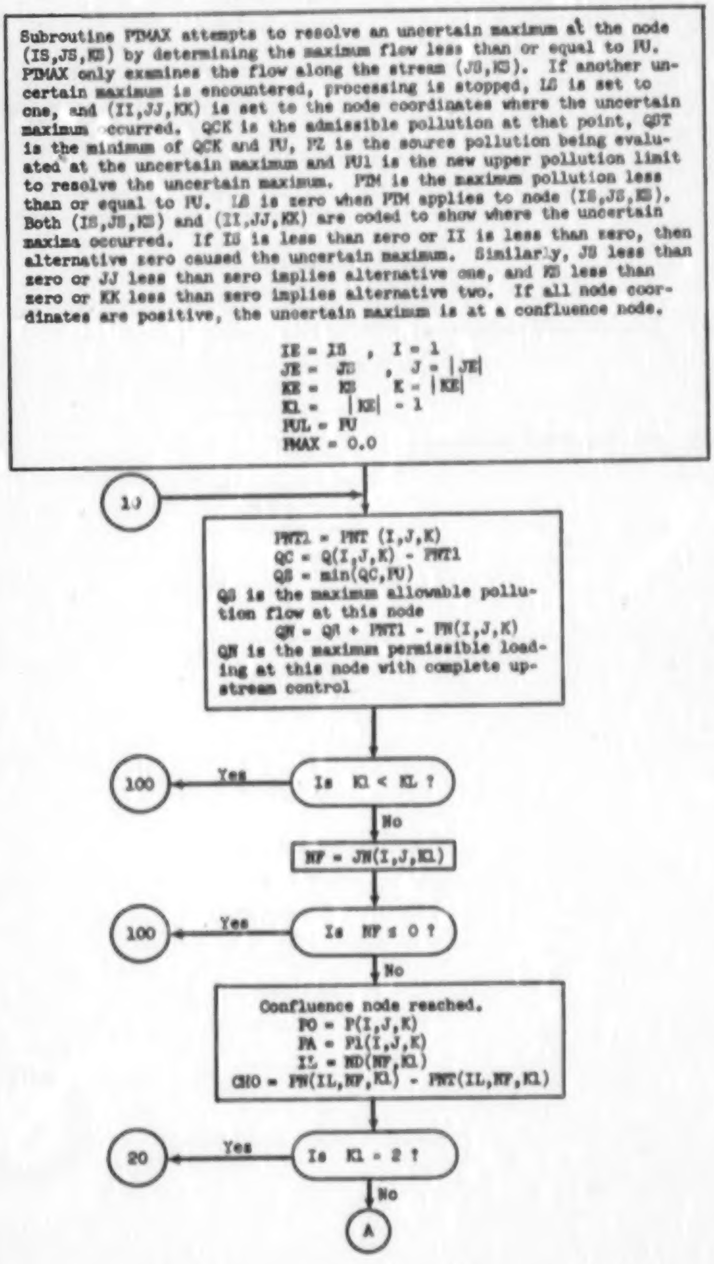

15
Increment to the second level stream
II = NFN(JJ,KK)
JJ = NFS(JJ)
Stream levels are now the same
20
25
No
Is JJ = JS ?
Yes
25
Yes
Is II > IS ?
No
NON = 1
Return
25
NON = 0
Return

Figure C.18 Function NON

Subroutine PTMAX(IS,JS,KS,II,JJ,KK,LS,PU,PTM,
PTMO, QST,QCK,QSN,PZ,PU1)



Subroutine PTMAX attempts to resolve an uncertain maximum at the node (IS,JS,KS) by determining the maximum flow less than or equal to PU. PTMAX only examines the flow along the stream (JS,KS). If another uncertain maximum is encountered, processing is stopped, LS is set to one, and (II,JJ,KK) is set to the node coordinates where the uncertain maximum occurred. QCK is the admissible pollution at that point, QST is the minimum of QCK and PU, PZ is the source pollution being evaluated at the uncertain maximum and PU1 is the new upper pollution limit to resolve the uncertain maximum. PTM is the maximum pollution less than or equal to PU. LS is zero when PTM applies to node (IS,JS,KS). Both (IS,JS,KS) and (II,JJ,KK) are coded to show where the uncertain maxima occurred. If IS is less than zero or II is less than zero, then alternative zero caused the uncertain maximum. Similarly, JS less than zero or JJ less than zero implies alternative one, and KS less than zero or KK less than zero implies alternative two. If all node coordinates are positive, the uncertain maximum is at a confluence node.

IE = IS , I = 1
JE = JS , J = |JE|
KE = KS K = |KE|
K1 = |KE| - 1
PUL = PU
PMAX = 0.0

PNT1 = PNT (I,J,K)
QC = Q(I,J,K) - PNT1
QS = min(QC,PU)
QS is the maximum allowable pollution flow at this node
QN = QS + PNT1 - PN(I,J,K)
QN is the maximum permissible loading at this node with complete upstream control

Is K1 < KL ?

NF = JN(I,J,K1)

Is NF ≤ 0 ?

Confluence node reached.
PO = P(I,J,K)
PA = P1(I,J,K)
IL = ND(NF,K1)
CNO = PN(IL,NF,K1) - PNT(IL,NF,K1)

Is K1 = 2 ?

Figure C.19 Subroutine PTMAX

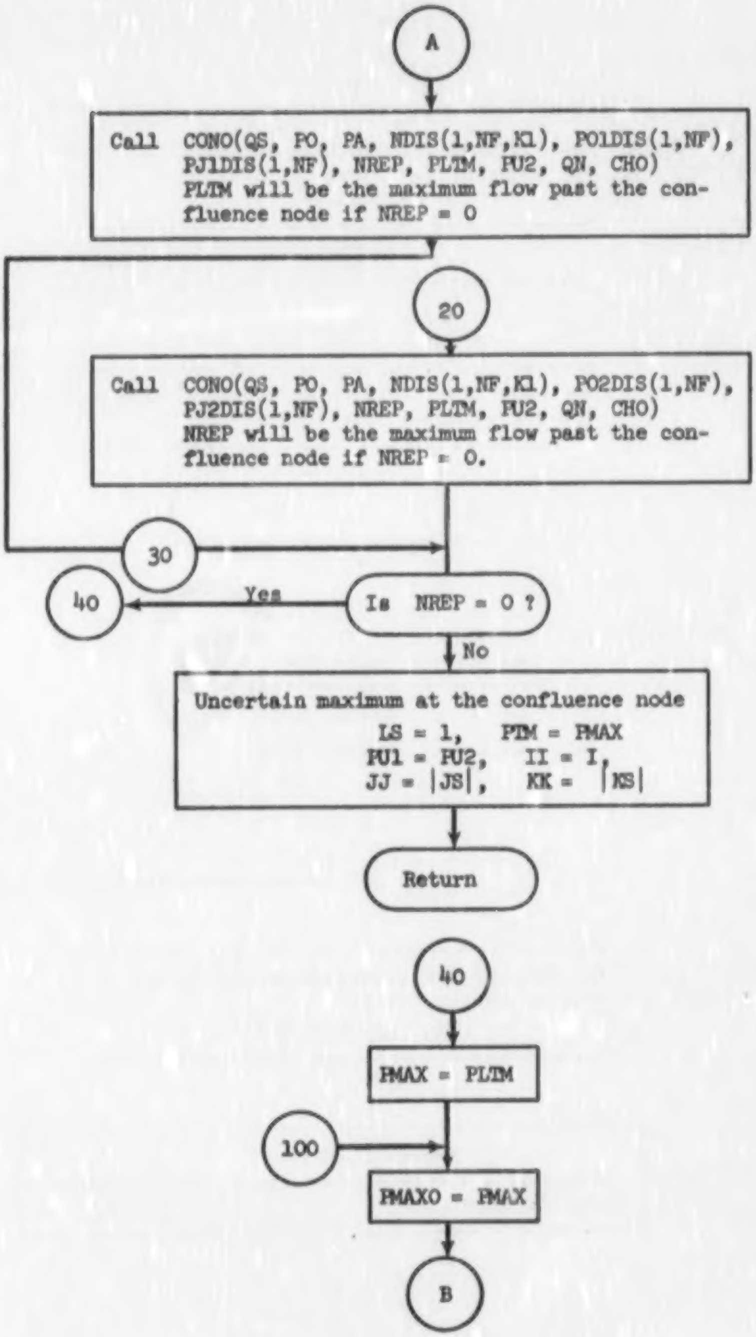

A
Call CONO(QS, PO, PA, NDIS(1,NF,K1), PO1DIS(1,NF), PJ1DIS(1,NF), NREP, PLTM, PU2, QN, CHO)
PLTM will be the maximum flow past the confluence node if NREP = 0
20
Call CONO(QS, PO, PA, NDIS(1,NF,K1), PO2DIS(1,NF), PJ2DIS(1,NF), NREP, PLTM, PU2, QN, CHO)
NREP will be the maximum flow past the confluence node if NREP = 0.
30
40
Yes
Is NREP = 0 ?
No
Uncertain maximum at the confluence node
LS = 1, PTM = PMAX
PU1 = PU2, II = I,
JJ = |JS|, KK = |KS|
Return
40
PMAX = PLTM
100
PMAXO = PMAX
B

Figure C.19 Subroutine PTMAX

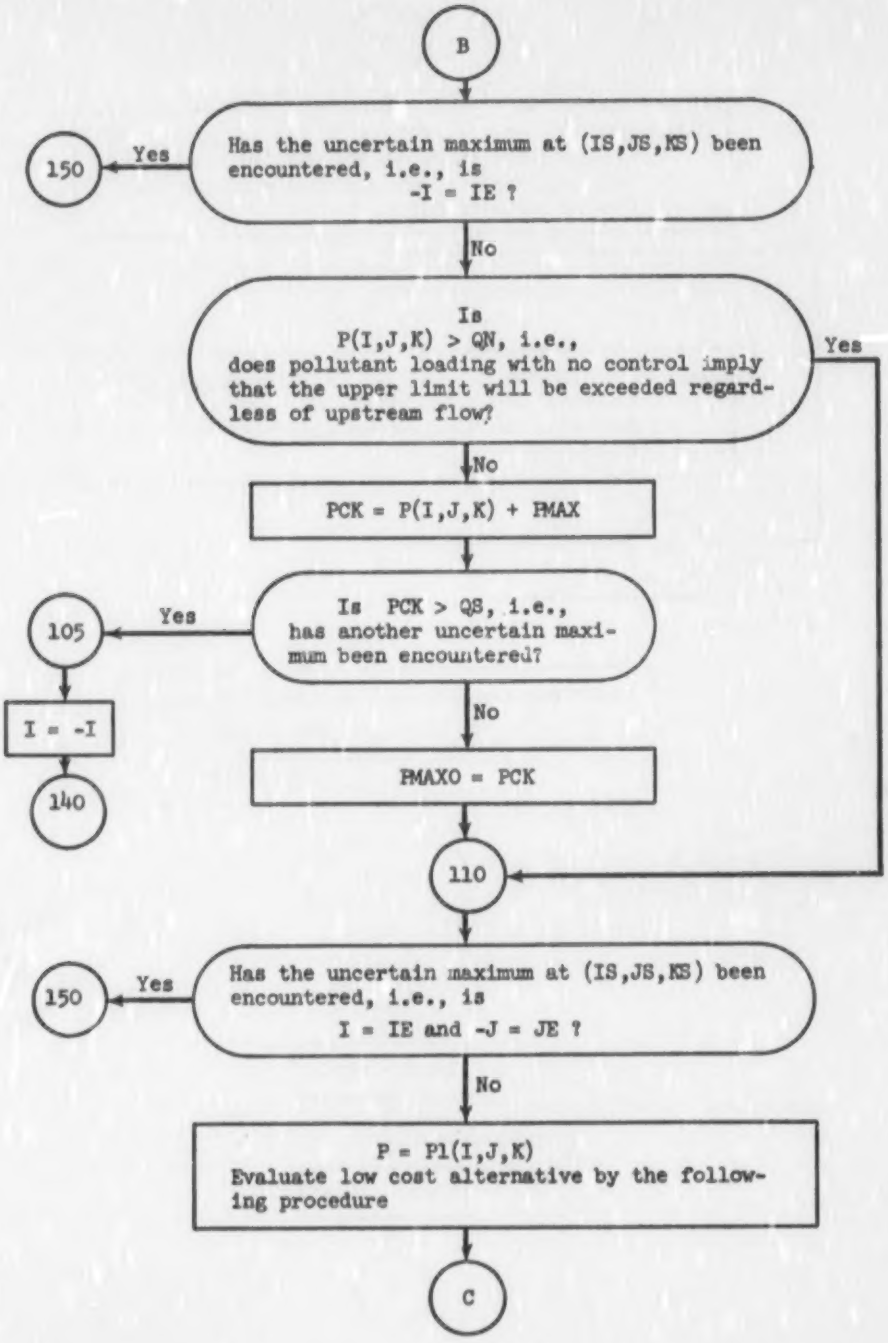

B
150
Yes
Has the uncertain maximum at (IS,JS,KS) been encountered, i.e., is
-I = IE ?
No
Is
P(I,J,K) > QN, i.e.,
does pollutant loading with no control imply that the upper limit will be exceeded regardless of upstream flow?
Yes
No
PCK = P(I,J,K) + PTMAX
105
Yes
Is PCK > QS, i.e.,
has another uncertain maximum been encountered?
I = -I
140
No
PTMAXO = PCK
110
150
Yes
Has the uncertain maximum at (IS,JS,KS) been encountered, i.e., is
I = IE and -J = JE ?
No
P = P1(I,J,K)
Evaluate low cost alternative by the following procedure
C

Figure C.19 Subroutine PTMAX

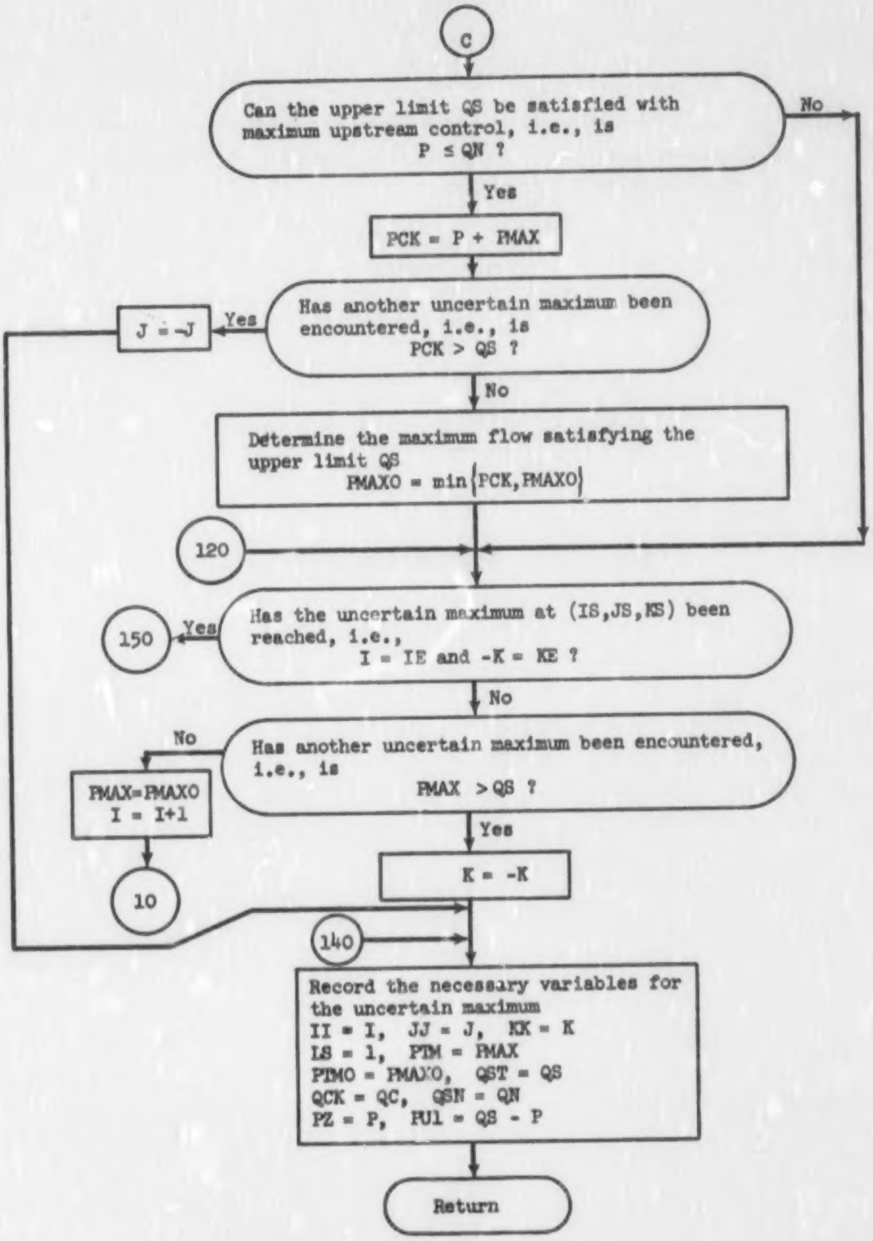

C
Can the upper limit QS be satisfied with maximum upstream control, i.e., is P ≤ QN ?
No
Yes
PCK = P + PMAX
Has another uncertain maximum been encountered, i.e., is PCK > QS ?
Yes
J = -J
No
Determine the maximum flow satisfying the upper limit QS
PMAXO = min{PCK,PMAXO}
120
Has the uncertain maximum at (IS,JS,KS) been reached, i.e., I = IE and -K = KE ?
Yes
150
No
Has another uncertain maximum been encountered, i.e., is PMAX > QS ?
No
PMAX=PMAXO
I = I+1
10
Yes
K = -K
140
Record the necessary variables for the uncertain maximum
II = I, JJ = J, KK = K
LS = 1, PTM = PMAX
PTMO = PMAXO, QST = QS
QCK = QC, QSN = QN
PZ = P, PU1 = QS - P
Return

Figure C.19 Subroutine PTMAX

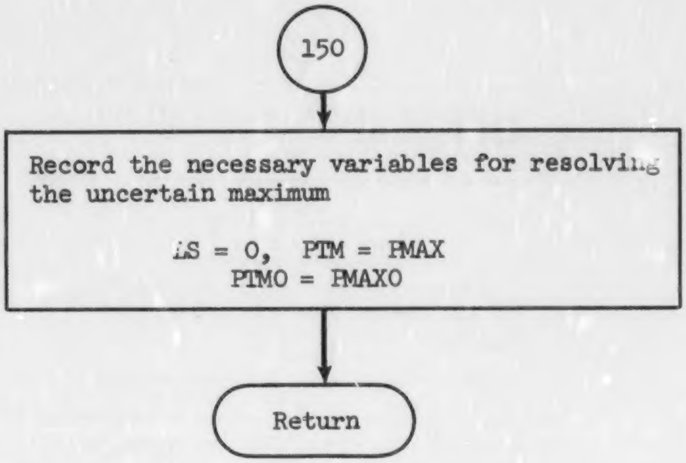

150
Record the necessary variables for resolving the uncertain maximum
LS = 0, PTM = PMAX
PTMO = PMAXO
Return

Figure C.19 Subroutine PTMAX

Subroutine PTMX(IJ,JJ,KK,PLTMAX,POUT)

PTMX computes the maximum pollution flow rate past node (II,JJ,KK) which is less than or equal to PLTMAX. PTMX resolves uncertain maxima.

KCN(J,K) = Value of KFN for confluence node being fed by stream (J,K). KCN(J,K) = 0 if admissible pollution values for the confluence node are not being calculated.

$$\text{KFN} = \begin{cases} \text{1 if admissible pollution flow to a confluence node from a lower level stream is being calculated.} \\ \text{2 if admissible pollution flow to a confluence node from the node's stream is being calculated.} \\ \text{0 if pollution flow to a node that is not a confluence node is being calculated.} \end{cases}$$

MDIS(I,K) = Maximum number of pollutant flows which can be stored for confluence nodes receiving flow from level K streams. Set I = 1 for output from level K stream, I = 2 for upstream output on level K+1 stream receiving flow from level K streams.

MU = Number of entries on the uncertain maximum list.

(MUI(I),MUJ(I),MUK(I)) = Objective node coordinates for the Ith entry on the uncertain maximum list.

NDIS(I,J,K) = Number of admissible pollutant flows calculated for confluence node receiving flow from level K stream J. Set I = 1 for level K stream J output and I = 2 for upstream output on level K+1 stream receiving flow from level K stream J.

PJ1DIS(I,J) = Ith admissible output pollutant flow just upstream from confluence node receiving flow from level 1 stream J. (Note that PJ1DIS(I,J) ≥ PJ1DIS(I+1,J).

PJ2DIS(I,J) = Level 2 analogue of PJ1DIS(I,J).

PMAL(I) = Maximum pollution flow rate at the time the i^{th} entry on uncertain maximum list was encountered.

PMAX = Maximum flow rate less than or equal to PLTMAX. (PMAX is used in determining POUT).

POUT = Pollution flow rate returned as maximum value at node (II,JJ,KK) less than or equal to PLTMAX.

PO1DIS(I,J) = Ith admissible output pollutant flow from level 1 stream J. (Note that PO1DIS(I,J,) is greater than or equal to PO1DIS(I+1,J).)

PO2DIS(I,J) = Level 2 stream analogue of PO1DIS(I,J).

PTM = Maximum pollution flow rate less than or equal to PU. (PTM is used in resolving uncertain maxima).

PU = Upper limit on pollution flow rate currently being used.

PUM(I) = Upper limit on pollution flow rate at time i^{th} entry on uncertain maximum list was encountered.

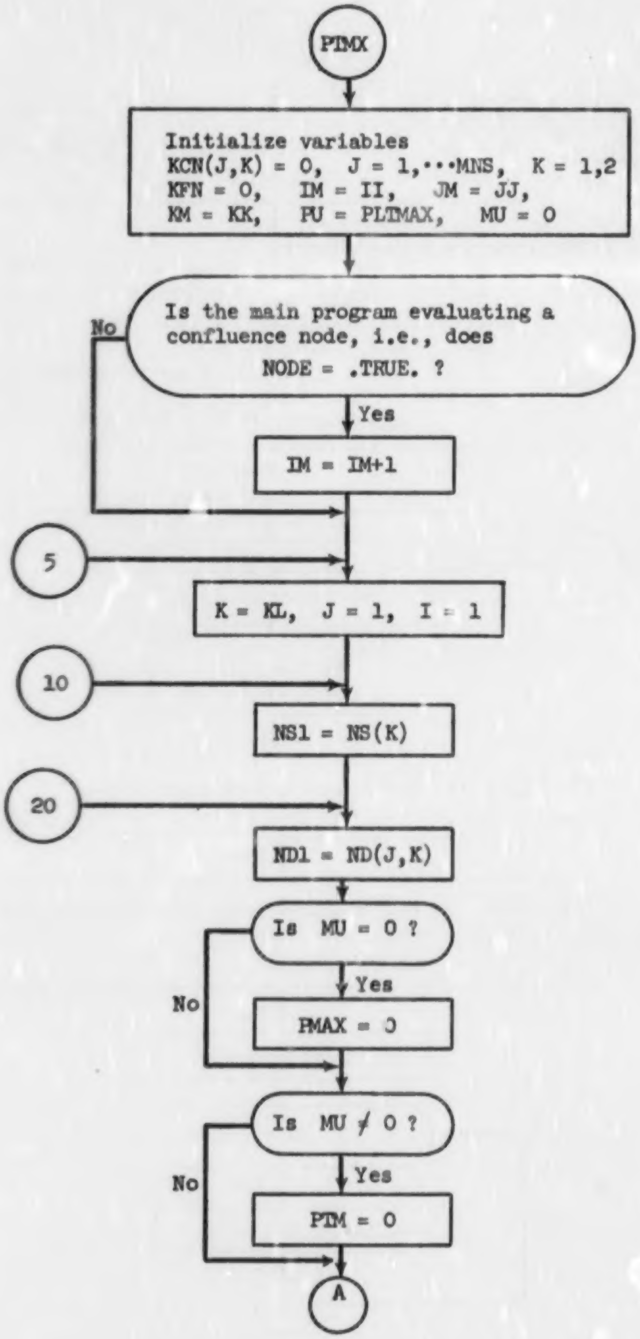

PTMX
Initialize variables
KCN(J,K) = 0, J = 1,•••MNS, K = 1,2
KFN = 0, IM = II, JM = JJ,
KM = KK, PU = PLTMAX, MU = 0
Is the main program evaluating a confluence node, i.e., does
NODE = .TRUE. ?
No
Yes
IM = IM+1
5
K = KL, J = 1, I = 1
10
NS1 = NS(K)
20
ND1 = ND(J,K)
Is MU = 0 ?
Yes
No
PMAX = 0
Is MU ≠ 0 ?
Yes
No
PTM = 0
A

Figure C.20 Subroutine PTMX

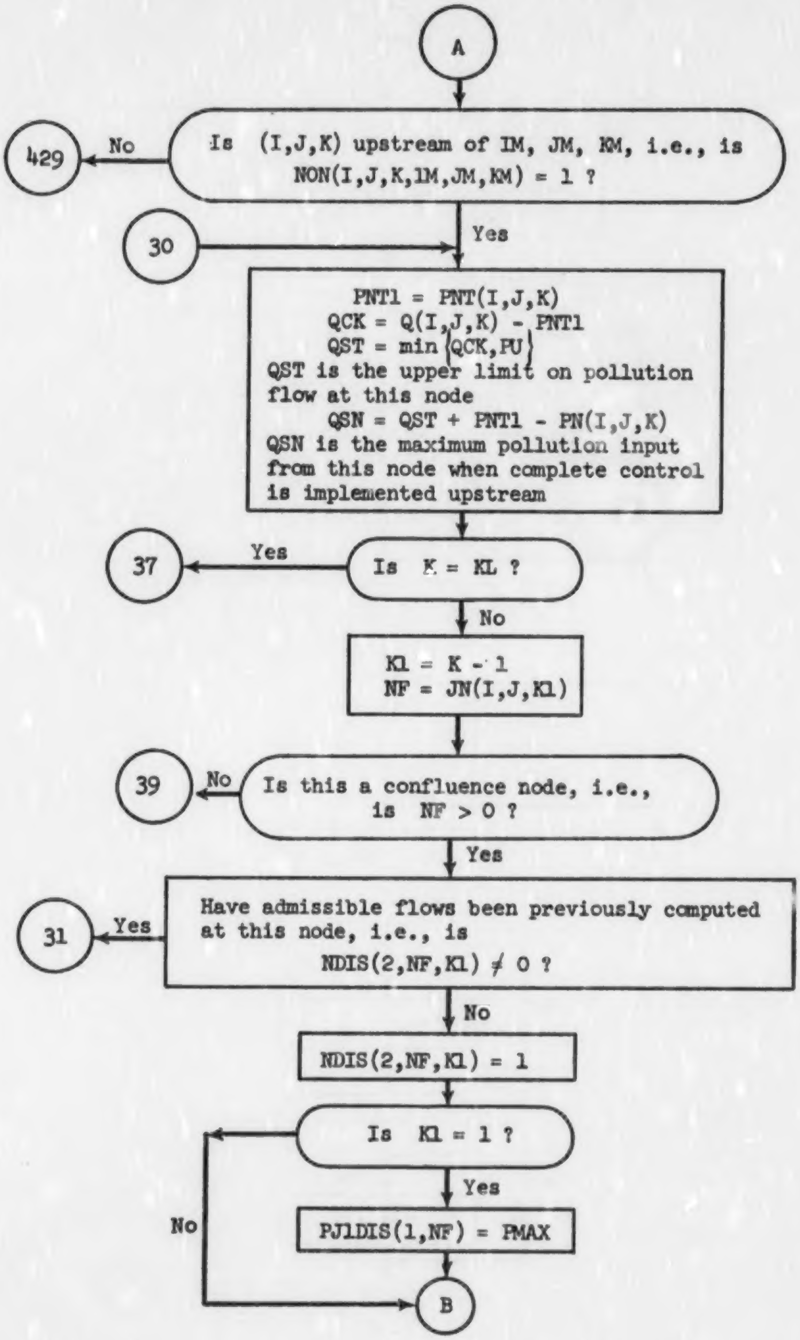

A
Is (I,J,K) upstream of IM, JM, KM, i.e., is NON(I,J,K,IM,JM,KM) = 1 ?
No
429
Yes
30
PNT1 = PNT(I,J,K)
QCK = Q(I,J,K) - PNT1
QST = min {QCK,PU}
QST is the upper limit on pollution flow at this node
QSN = QST + PNT1 - PN(I,J,K)
QSN is the maximum pollution input from this node when complete control is implemented upstream
Is K = KL ?
Yes
37
No
K1 = K - 1
NF = JN(I,J,K1)
Is this a confluence node, i.e., is NF > 0 ?
No
39
Yes
Have admissible flows been previously computed at this node, i.e., is NDIS(2,NF,K1) ≠ 0 ?
Yes
31
No
NDIS(2,NF,K1) = 1
Is K1 = 1 ?
Yes
PJ1DIS(1,NF) = PMAX
No
B

Figure C.20 Subroutine PIMX

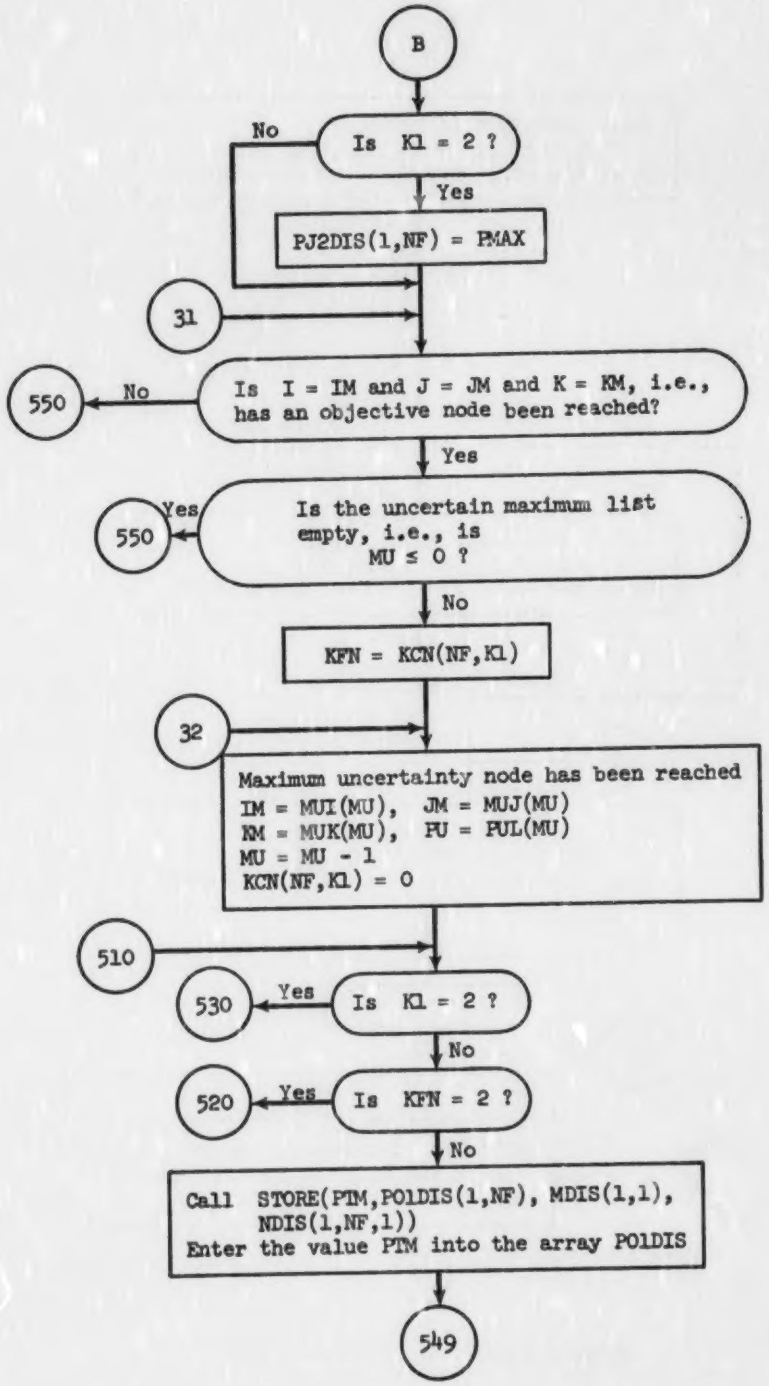

B
Is K1 = 2 ?
No
Yes
PJ2DIS(1,NF) = PMAX
31
Is I = IM and J = JM and K = KM, i.e., has an objective node been reached?
550
No
Yes
Is the uncertain maximum list empty, i.e., is MU ≤ 0 ?
550
Yes
No
KFN = KCN(NF,K1)
32
Maximum uncertainty node has been reached
IM = MUI(MU), JM = MUJ(MU)
KM = MUK(MU), PU = PUL(MU)
MU = MU - 1
KCN(NF,K1) = 0
510
Is K1 = 2 ?
530
Yes
No
Is KFN = 2 ?
520
Yes
No
Call STORE(PTM,P01DIS(1,NF), MDIS(1,1), NDIS(1,NF,1))
Enter the value PTM into the array P01DIS
549

Figure C.20 Subroutine PTMX

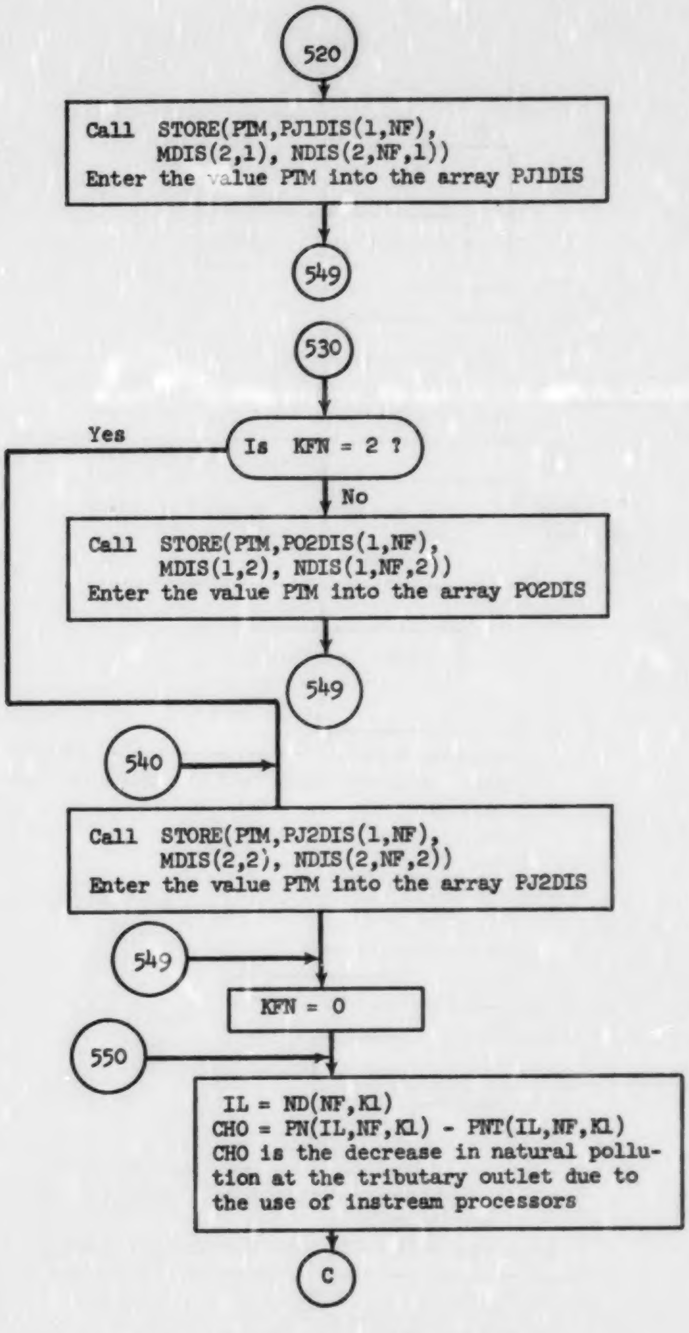

520
Call STORE(PTM,PJ1DIS(1,NF),
MDIS(2,1), NDIS(2,NF,1))
Enter the value PTM into the array PJ1DIS
549
530
Yes
Is KFN = 2 ?
No
Call STORE(PTM,PO2DIS(1,NF),
MDIS(1,2), NDIS(1,NF,2))
Enter the value PTM into the array PO2DIS
549
540
Call STORE(PTM,PJ2DIS(1,NF),
MDIS(2,2), NDIS(2,NF,2))
Enter the value PTM into the array PJ2DIS
549
KFN = 0
550
IL = ND(NF,K1)
CHO = PN(IL,NF,K1) - PNT(IL,NF,K1)
CHO is the decrease in natural pollution at the tributary outlet due to the use of instream processors
C

Figure C.20 Subroutine PTMX

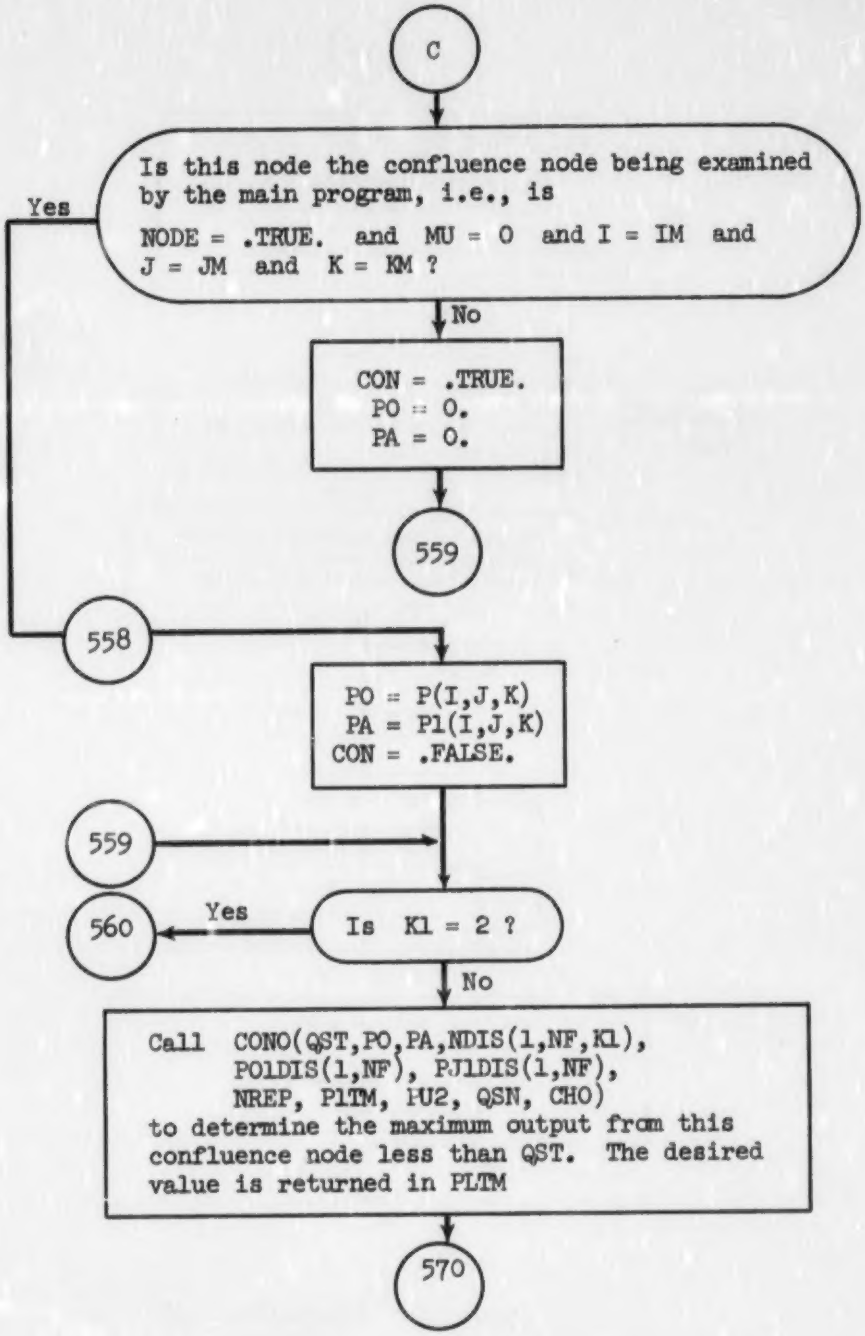

C
Is this node the confluence node being examined by the main program, i.e., is NODE = .TRUE. and MU = 0 and I = IM and J = JM and K = KM ?
Yes
No
CON = .TRUE.
PO = 0.
PA = 0.
559
558
PO = P(I,J,K)
PA = P1(I,J,K)
CON = .FALSE.
559
Is K1 = 2 ?
Yes
560
No
Call CONO(QST,PO,PA,NDIS(1,NF,K1), PO1DIS(1,NF), PJ1DIS(1,NF), NREP, P1TM, HU2, QSN, CHO) to determine the maximum output from this confluence node less than QST. The desired value is returned in PLTM
570

Figure C.20 Subroutine PTMX

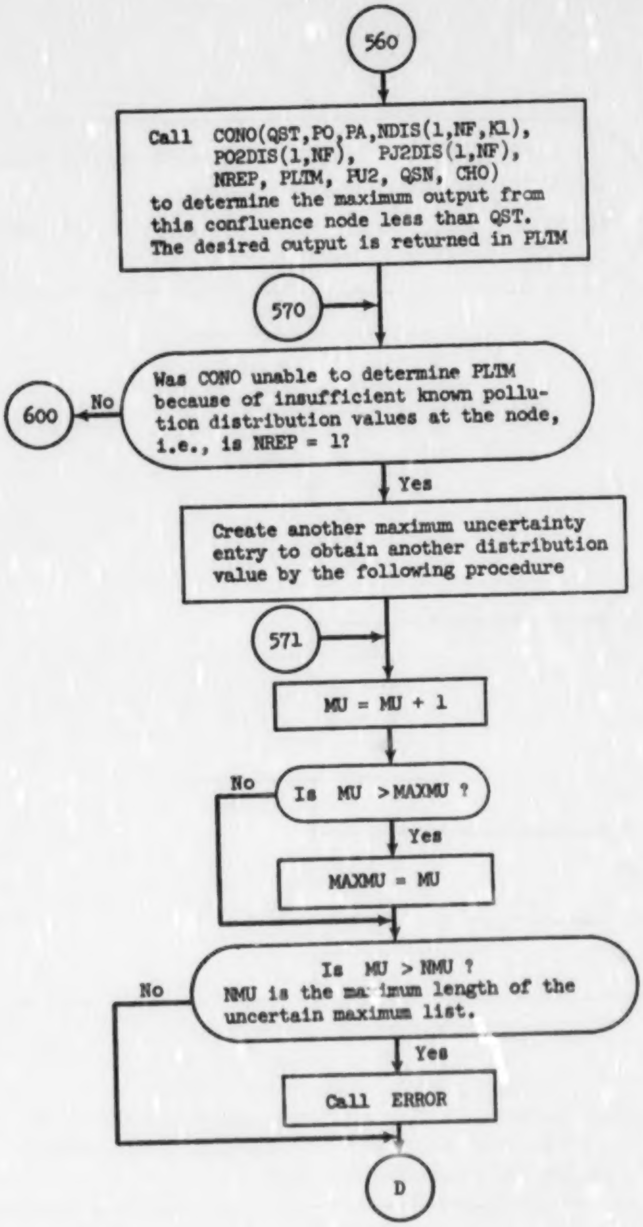

560
Call CONO(QST,PO,PA,NDIS(1,NF,K1), PO2DIS(1,NF), PJ2DIS(1,NF), NREP, PLTM, PU2, QSN, CHO) to determine the maximum output from this confluence node less than QST. The desired output is returned in PLTM
570
Was CONO unable to determine PLTM because of insufficient known pollution distribution values at the node, i.e., is NREP = 1?
No
600
Yes
Create another maximum uncertainty entry to obtain another distribution value by the following procedure
571
MU = MU + 1
Is MU >MAXMU ?
No
Yes
MAXMU = MU
Is MU > NMU ? NMU is the maximum length of the uncertain maximum list.
No
Yes
Call ERROR
D

Figure C.20 Subroutine PTMX

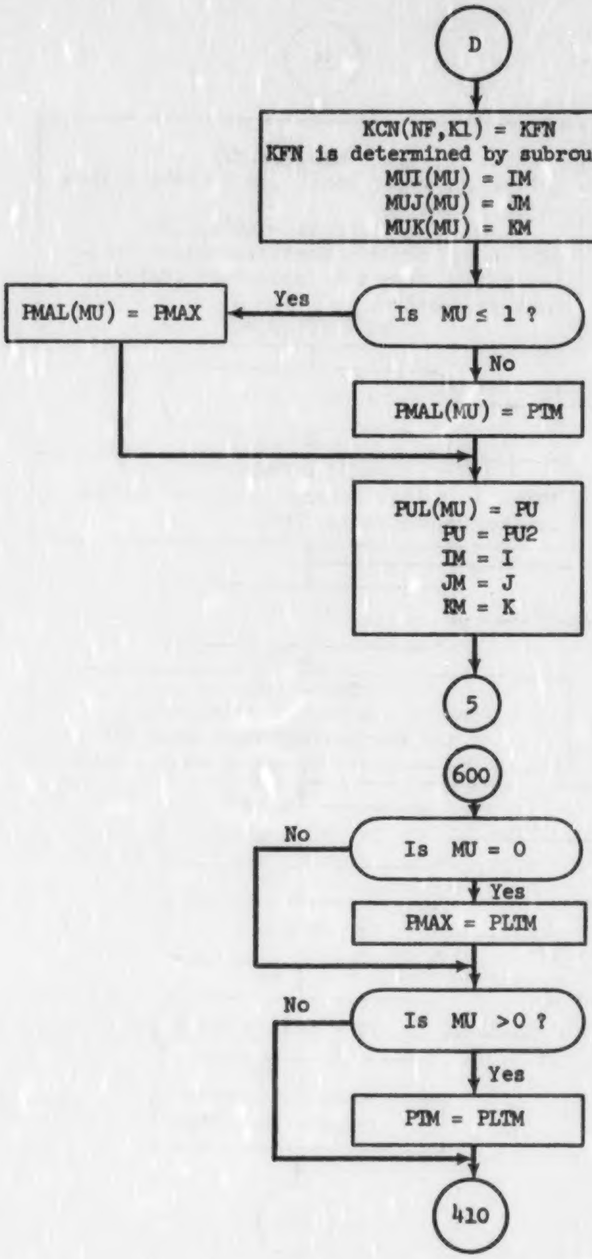

D
KCN(NF,K1) = KFN
KFN is determined by subroutine CONO
MUI(MU) = IM
MUJ(MU) = JM
MUK(MU) = KM
Is MU ≤ 1 ?
Yes
PMAL(MU) = PMAX
No
PMAL(MU) = PTM
PUL(MU) = PU
PU = PU2
IM = I
JM = J
KM = K
5
600
Is MU = 0
No
Yes
PMAX = PLTM
Is MU >0 ?
No
Yes
PTM = PLTM
410

Figure C.20 Subroutine PTMX

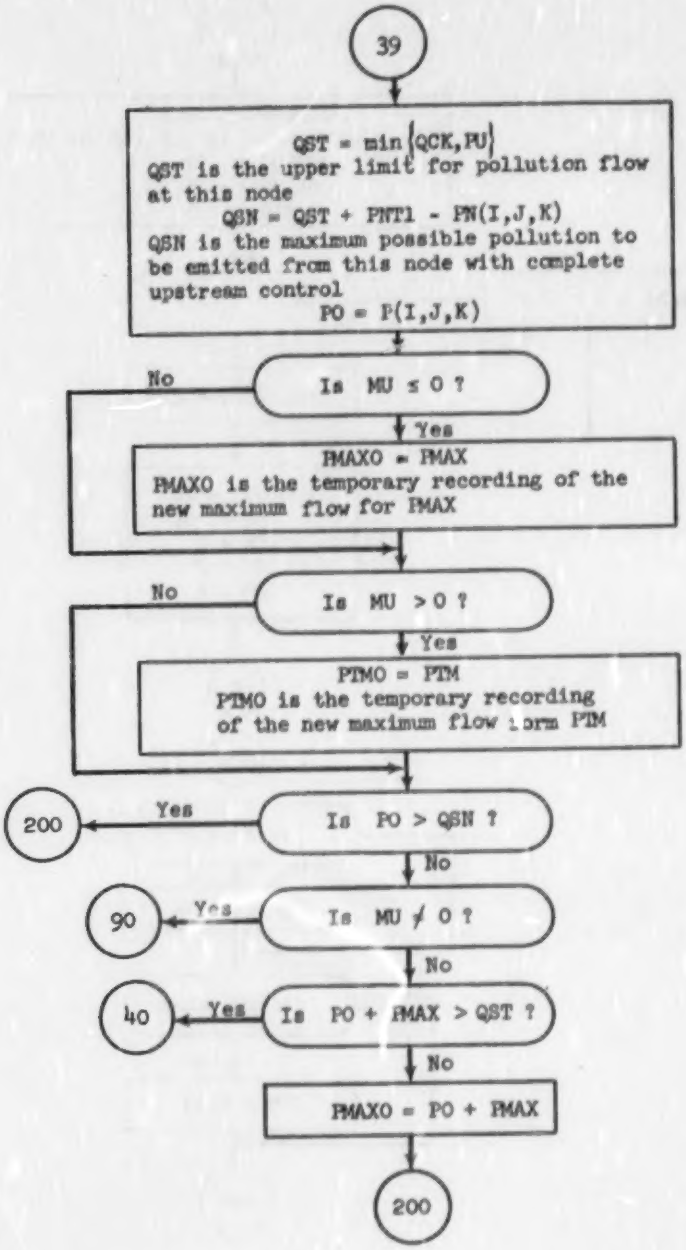

39
QST = min{QCK,PU}
QST is the upper limit for pollution flow at this node
QSN = QST + PNT1 - PN(I,J,K)
QSN is the maximum possible pollution to be emitted from this node with complete upstream control
PO = P(I,J,K)
Is MU ≤ 0 ?
No
Yes
PMAXO = PMAX
PMAXO is the temporary recording of the new maximum flow for PMAX
Is MU > 0 ?
No
Yes
PTMO = PTM
PTMO is the temporary recording of the new maximum flow form PTM
Is PO > QSN ?
Yes
200
No
Is MU ≠ 0 ?
Yes
90
No
Is PO + PMAX > QST ?
Yes
40
No
PMAXO = PO + PMAX
200

Figure C.20 Subroutine PTMX

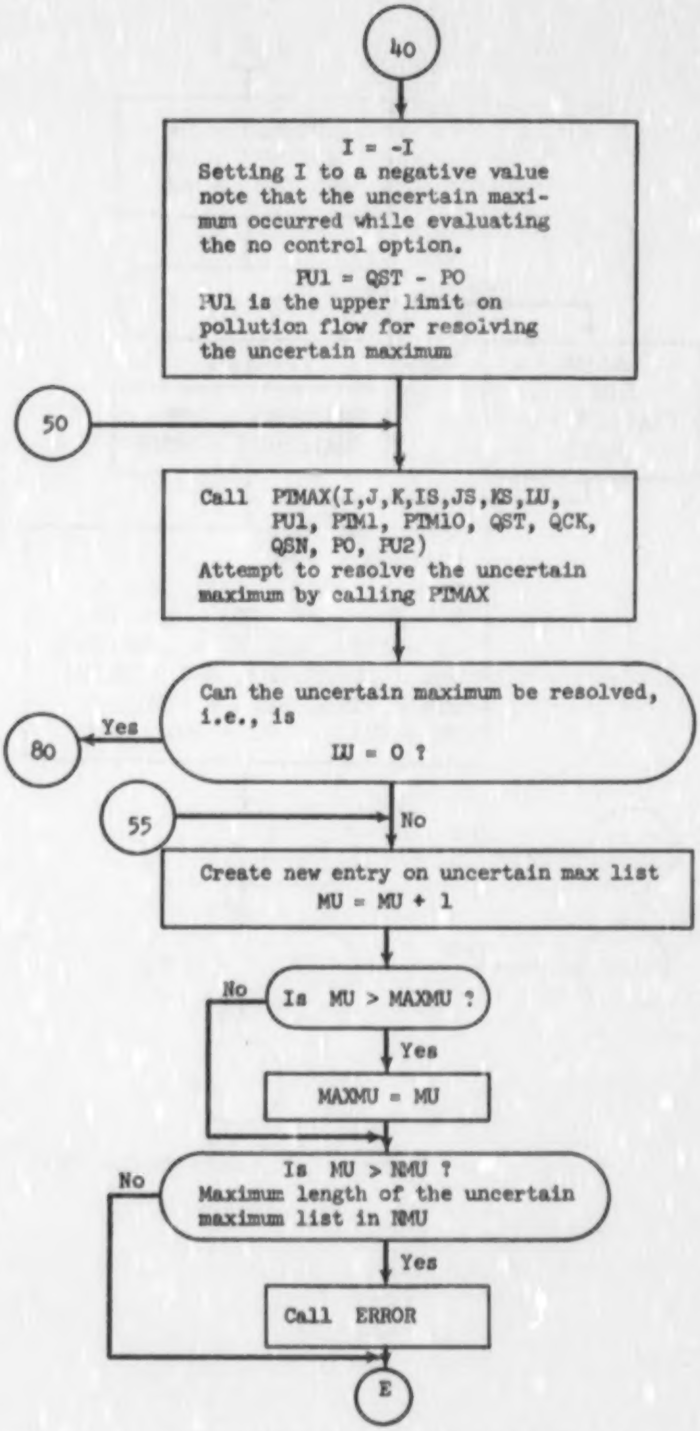

40
I = -I
Setting I to a negative value note that the uncertain maximum occurred while evaluating the no control option.
PU1 = QST - PO
PU1 is the upper limit on pollution flow for resolving the uncertain maximum
50
Call PTMAX(I,J,K,IS,JS,KS,LU,
PU1, PTM1, PTM10, QST, QCK,
QSN, PO, PU2)
Attempt to resolve the uncertain maximum by calling PTMAX
Can the uncertain maximum be resolved, i.e., is
LU = 0 ?
Yes
80
55
No
Create new entry on uncertain max list
MU = MU + 1
No
Is MU > MAXMU ?
Yes
MAXMU = MU
No
Is MU > NMU ?
Maximum length of the uncertain maximum list in NMU
Yes
Call ERROR
E

Figure C.20 Subroutine PTMX

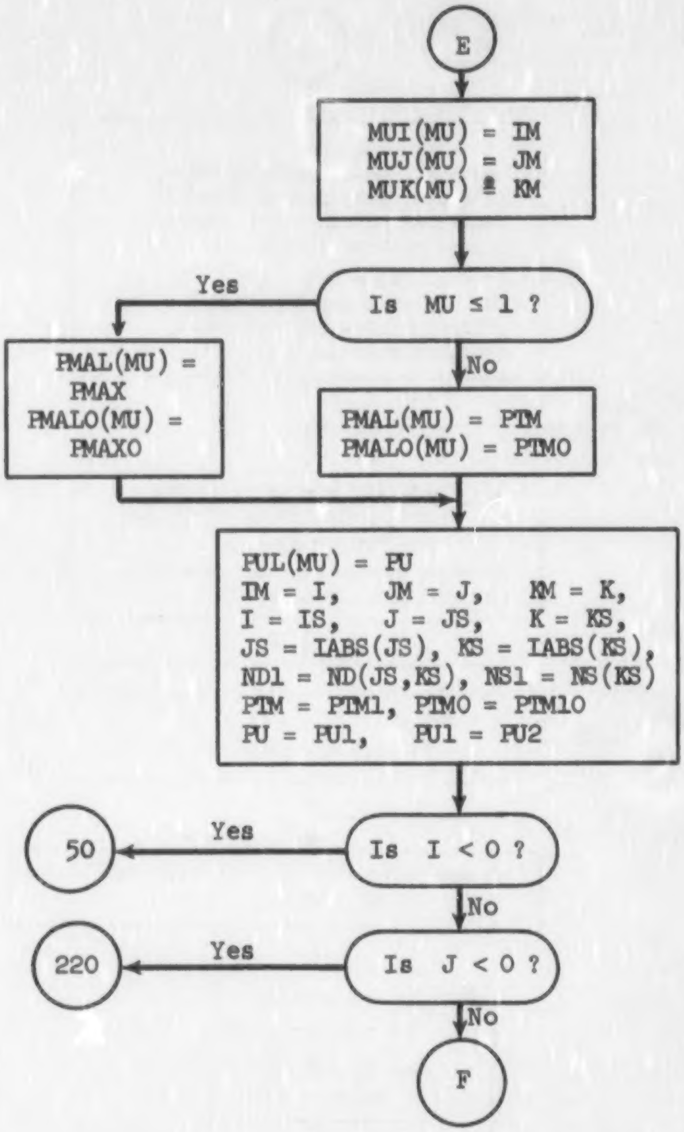

E
MUI(MU) = IM
MUJ(MU) = JM
MUK(MU) = KM
Is MU ≤ 1 ?
Yes
PMAL(MU) =
PMAX
PMALO(MU) =
PMAXO
No
PMAL(MU) = PTM
PMALO(MU) = PTMO
PUL(MU) = PU
IM = I, JM = J, KM = K,
I = IS, J = JS, K = KS,
JS = IABS(JS), KS = IABS(KS),
ND1 = ND(JS,KS), NS1 = NS(KS)
PTM = PTM1, PTMO = PTM10
PU = PU1, PU1 = PU2
Is I < 0 ?
Yes
50
No
Is J < 0 ?
Yes
220
No
F

Figure C.20 Subroutine PTMX

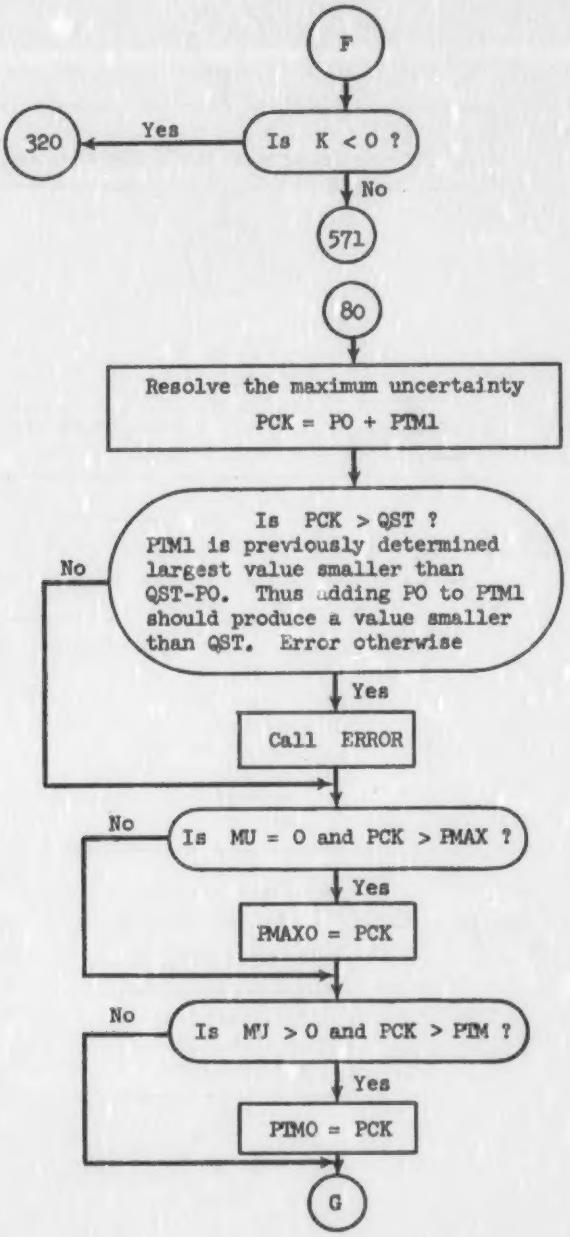

F
320
Yes
Is K < 0 ?
No
571
80
Resolve the maximum uncertainty
PCK = PO + PTM1
Is PCK > QST ?
PTM1 is previously determined largest value smaller than QST-PO. Thus adding PO to PTM1 should produce a value smaller than QST. Error otherwise
No
Yes
Call ERROR
No
Is MU = 0 and PCK > PMAX ?
Yes
PMAXO = PCK
No
Is MJ > 0 and PCK > PTM ?
Yes
PTMO = PCK
G

Figure C.20 Subroutine PTMX

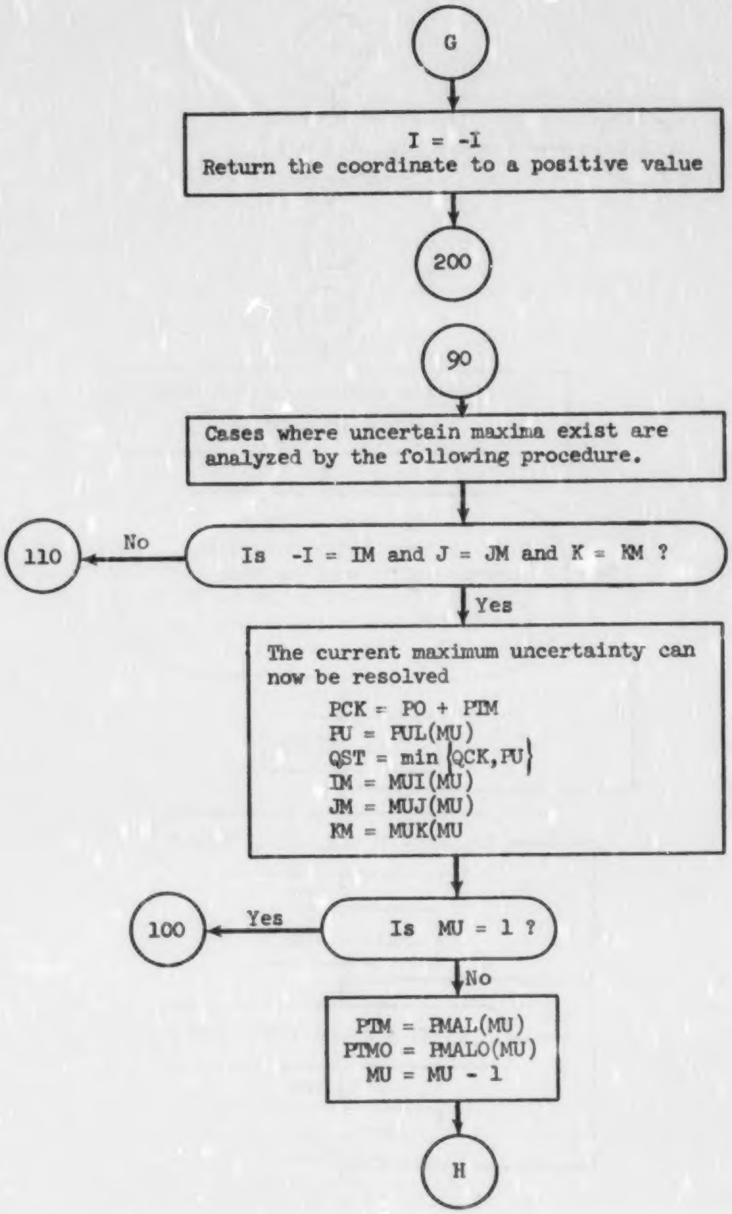

G
I = -I
Return the coordinate to a positive value
200
90
Cases where uncertain maxima exist are analyzed by the following procedure.
110
No
Is -I = IM and J = JM and K = KM ?
Yes
The current maximum uncertainty can now be resolved
PCK = PO + PTM
PU = PUL(MU)
QST = min{QCK,PU}
IM = MUI(MU)
JM = MUJ(MU)
KM = MUK(MU
100
Yes
Is MU = 1 ?
No
PTM = PMAL(MU)
PTMO = PMALO(MU)
MU = MU - 1
H

Figure C.20 Subroutine PTMX

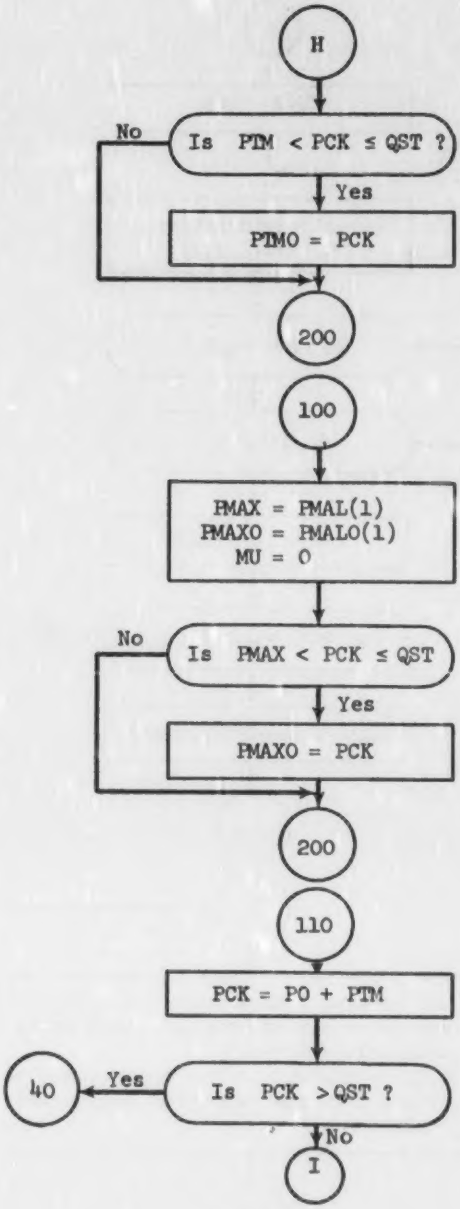

H
No
Is PTM < PCK ≤ QST ?
Yes
PTMO = PCK
200
100
PMAX = PMAL(1)
PMAXO = PMALO(1)
MU = 0
No
Is PMAX < PCK ≤ QST
Yes
PMAXO = PCK
200
110
PCK = PO + PTM
40
Yes
Is PCK >QST ?
No
I

Figure C.20 Subroutine PTMX

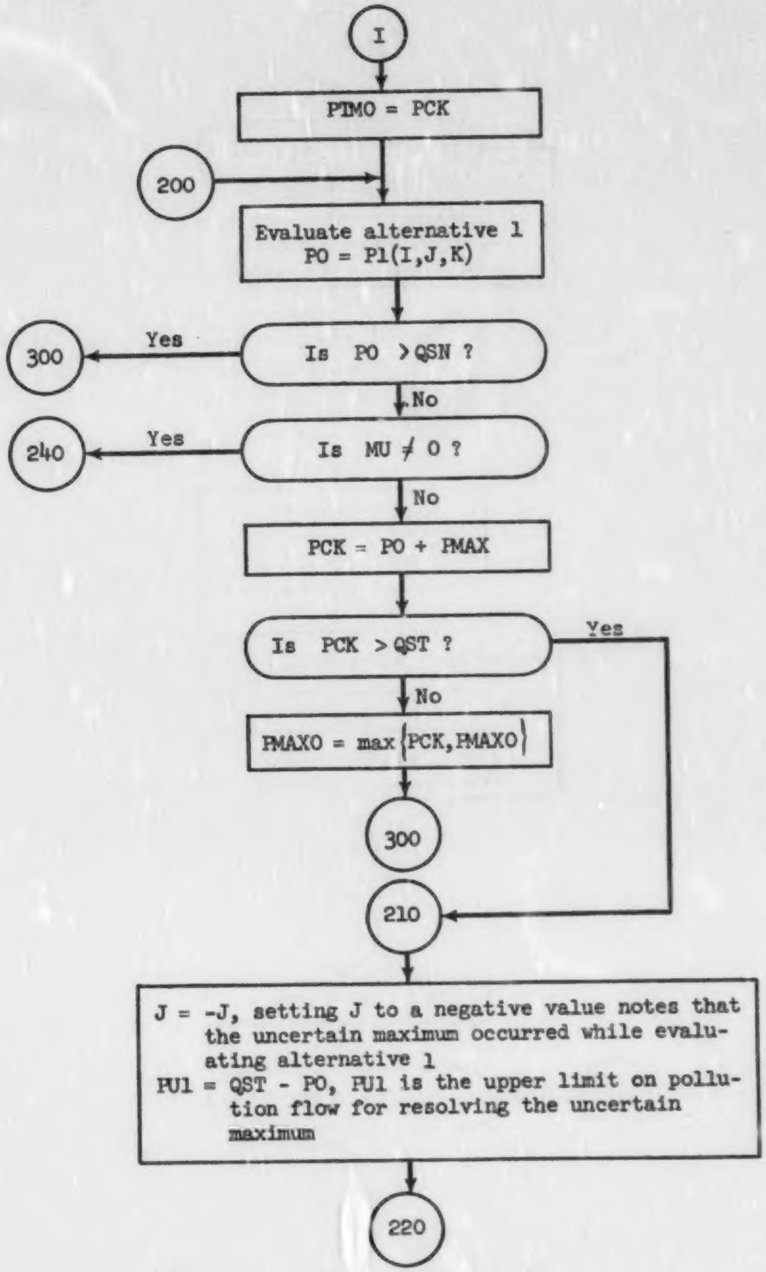

I
PTMO = PCK
200
Evaluate alternative 1
PO = P1(I,J,K)
300
Yes
Is PO >QSN ?
No
240
Yes
Is MU ≠ 0 ?
No
PCK = PO + PMAX
Is PCK >QST ?
Yes
No
PMAXO = max{PCK,PMAXO}
300
210
J = -J, setting J to a negative value notes that the uncertain maximum occurred while evaluating alternative 1
PU1 = QST - PO, PU1 is the upper limit on pollution flow for resolving the uncertain maximum
220

Figure C.20 Subroutine PTMX

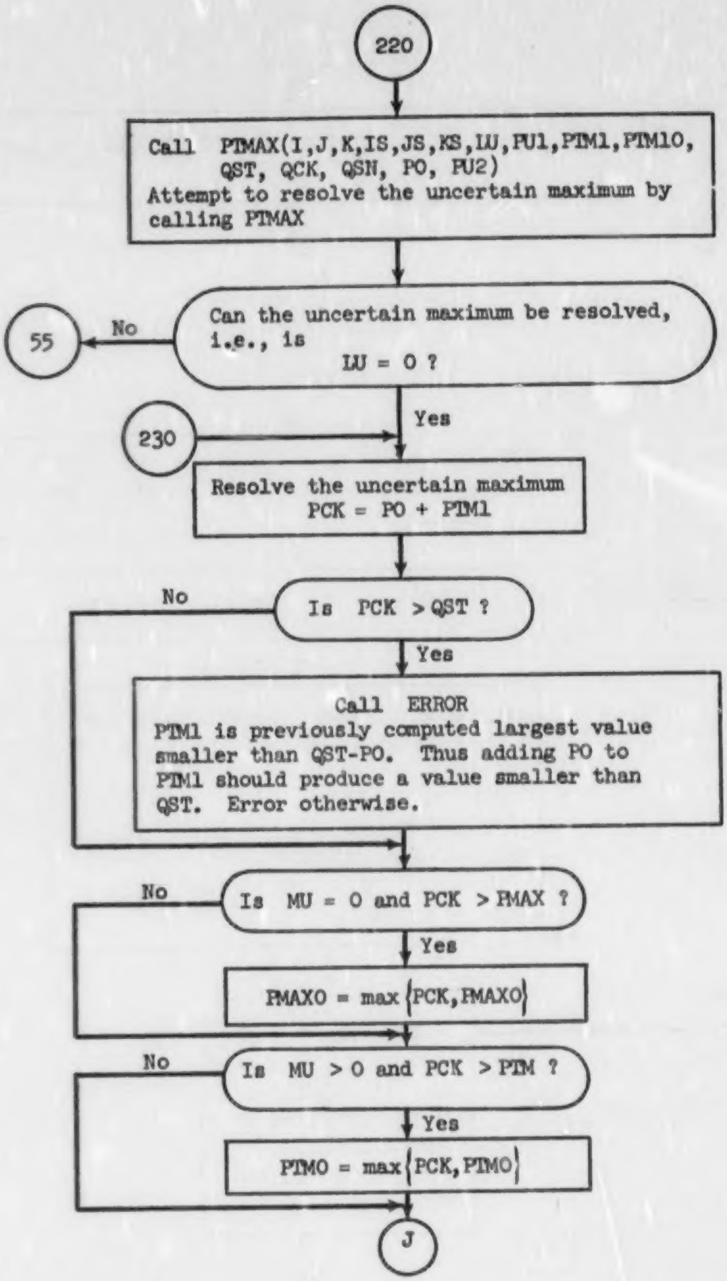

220
Call PTMAX(I,J,K,IS,JS,KS,LU,PU1,PTM1,PTM10, QST, QCK, QSN, PO, PU2)
Attempt to resolve the uncertain maximum by calling PTMAX
Can the uncertain maximum be resolved, i.e., is LU = 0 ?
No
55
Yes
230
Resolve the uncertain maximum PCK = PO + PTM1
Is PCK > QST ?
No
Yes
Call ERROR
PTM1 is previously computed largest value smaller than QST-PO. Thus adding PO to PTM1 should produce a value smaller than QST. Error otherwise.
Is MU = 0 and PCK > PMAX ?
No
Yes
PMAXO = max{PCK,PMAXO}
Is MU > 0 and PCK > PTM ?
No
Yes
PTMO = max{PCK,PTMO}
J

Figure C.20 Subroutine PTMX

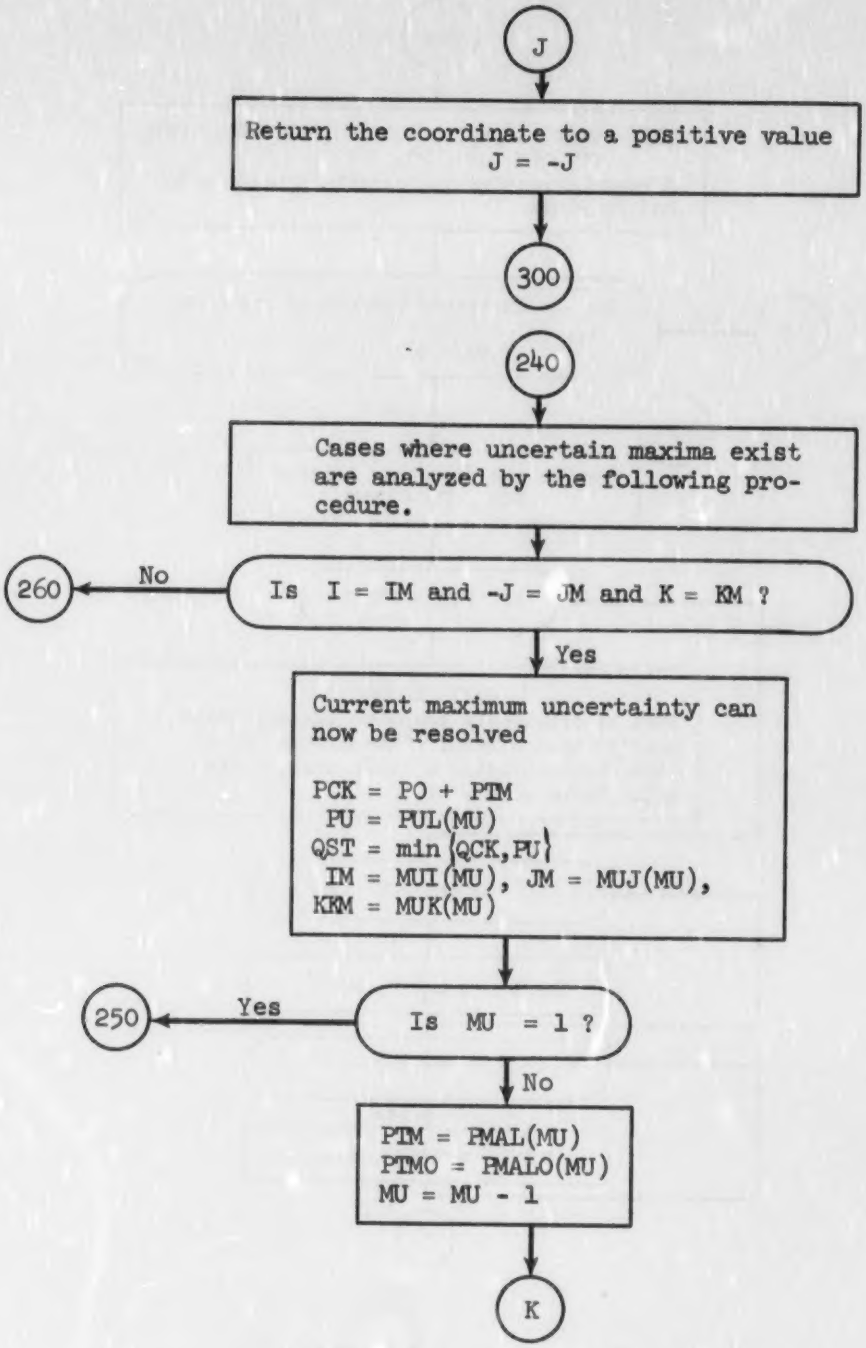

J
Return the coordinate to a positive value
J = -J
300
240
Cases where uncertain maxima exist are analyzed by the following procedure.
Is I = IM and -J = JM and K = KM ?
No
260
Yes
Current maximum uncertainty can now be resolved
PCK = PO + PTM
PU = PUL(MU)
QST = min {QCK,PU}
IM = MUI(MU), JM = MUJ(MU),
KKM = MUK(MU)
Is MU = 1 ?
Yes
250
No
PTM = PMAL(MU)
PTMO = PMALO(MU)
MU = MU - 1
K

Figure C.20 Subroutine PTMX

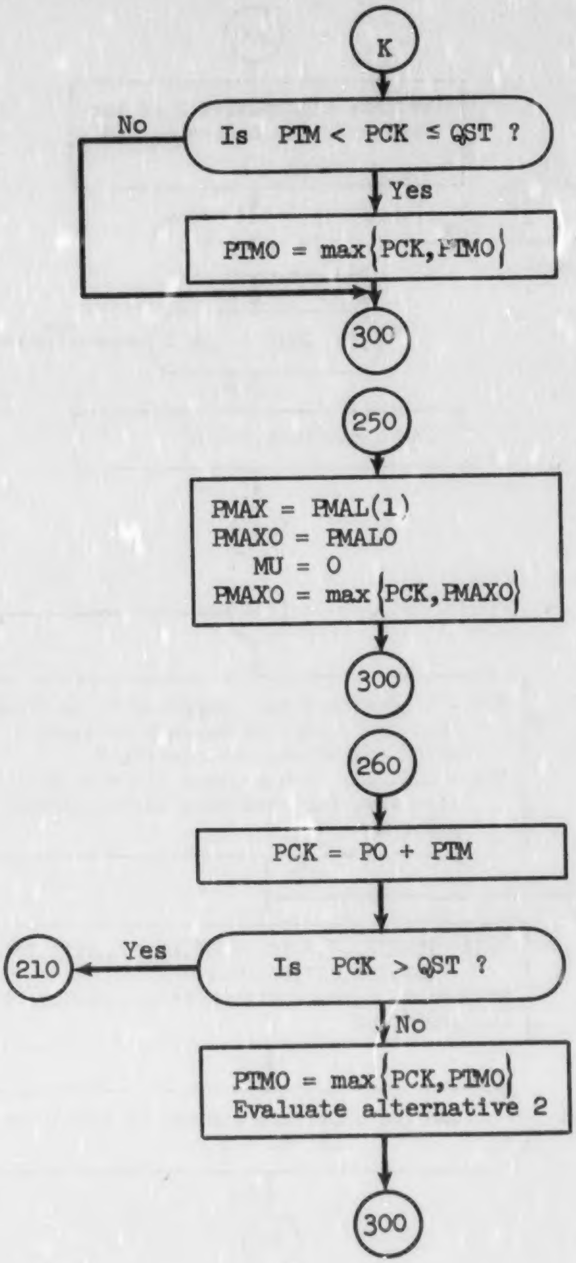

K
Is PTM < PCK ≤ QST ?
No
Yes
PTMO = max{PCK, PTMO}
300
250
PMAX = PMAL(1)
PMAXO = PMALO
MU = 0
PMAXO = max{PCK, PMAXO}
300
260
PCK = PO + PTM
Is PCK > QST ?
Yes
210
No
PTMO = max{PCK, PTMO}
Evaluate alternative 2
300

Figure C.20 Subroutine PTMX

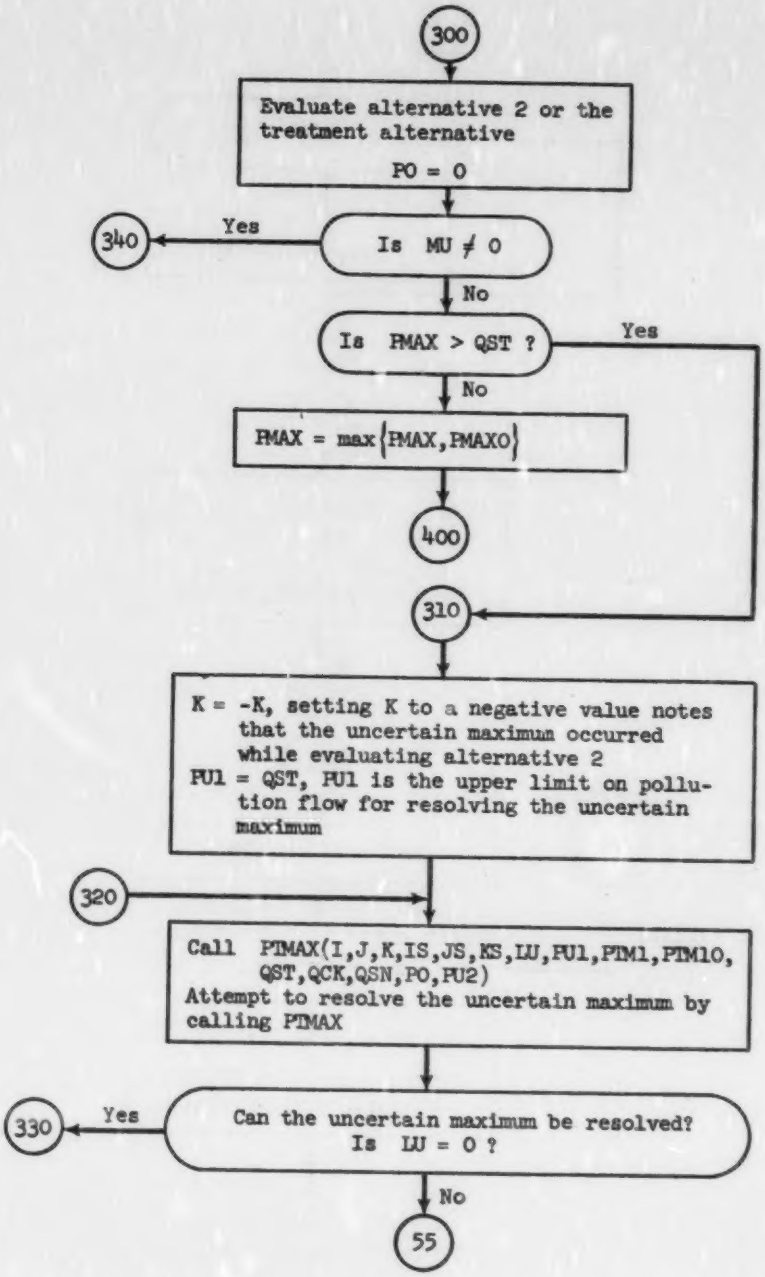

300
Evaluate alternative 2 or the treatment alternative
PO = 0
340
Yes
Is MU ≠ 0
No
Is PMAX > QST ?
Yes
No
PMAX = max{PMAX,PMAXO}
400
310
K = -K, setting K to a negative value notes that the uncertain maximum occurred while evaluating alternative 2
PU1 = QST, PU1 is the upper limit on pollution flow for resolving the uncertain maximum
320
Call PTMAX(I,J,K,IS,JS,KS,LU,PU1,PTM1,PTM10, QST,QCK,QSN,PO,PU2)
Attempt to resolve the uncertain maximum by calling PTMAX
330
Yes
Can the uncertain maximum be resolved? Is LU = 0 ?
No
55

Figure C.20 Subroutine PTMX

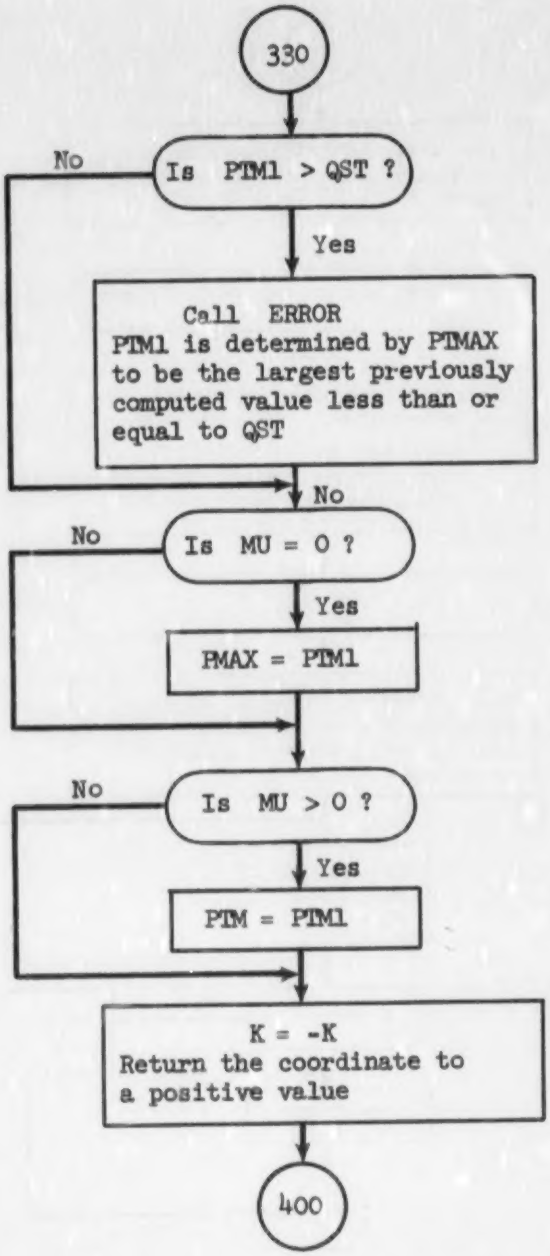

330
Is PTM1 > QST ?
No
Yes
Call ERROR
PTM1 is determined by PTMAX to be the largest previously computed value less than or equal to QST
No
Is MU = 0 ?
No
Yes
PMAX = PTM1
Is MU > 0 ?
No
Yes
PTM = PTM1
K = -K
Return the coordinate to a positive value
400

Figure C.20 Subroutine PTMX

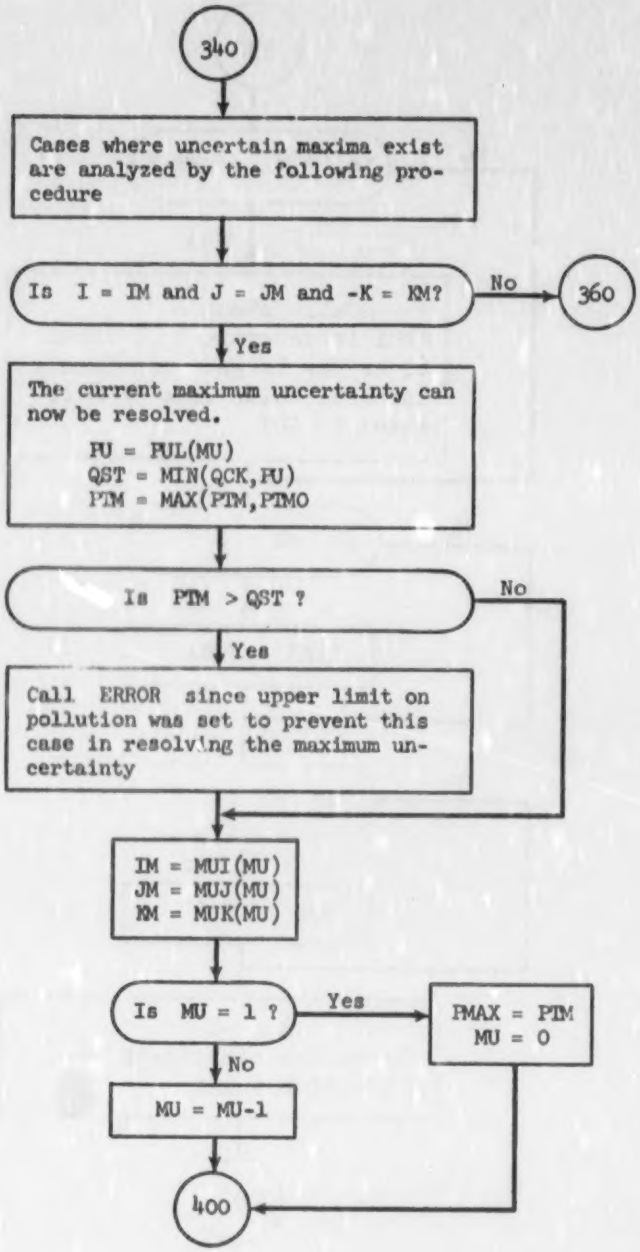

340
Cases where uncertain maxima exist are analyzed by the following procedure
Is I = IM and J = JM and -K = KM?
No
360
Yes
The current maximum uncertainty can now be resolved.
PU = PUL(MU)
QST = MIN(QCK,PU)
PTM = MAX(PTM,PTMO
Is PTM > QST ?
No
Yes
Call ERROR since upper limit on pollution was set to prevent this case in resolving the maximum uncertainty
IM = MUI(MU)
JM = MUJ(MU)
KM = MUK(MU)
Is MU = 1 ?
Yes
PMAX = PTM
MU = 0
No
MU = MU-1
400

Figure C.20 Subroutine PTMX

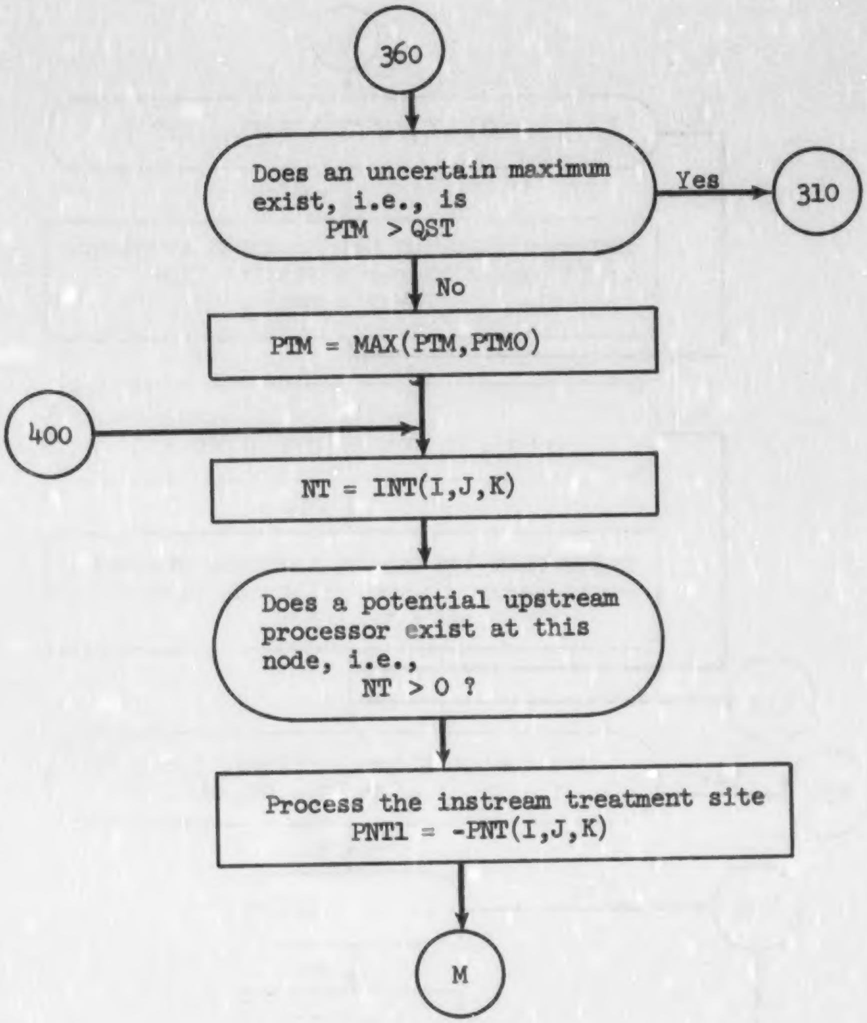

360
Does an uncertain maximum exist, i.e., is PTM > QST
Yes
310
No
PTM = MAX(PTM,PTMO)
400
NT = INT(I,J,K)
Does a potential upstream processor exist at this node, i.e., NT > 0 ?
Process the instream treatment site PNT1 = -PNT(I,J,K)
M

Figure C.20 Subroutine PTMX

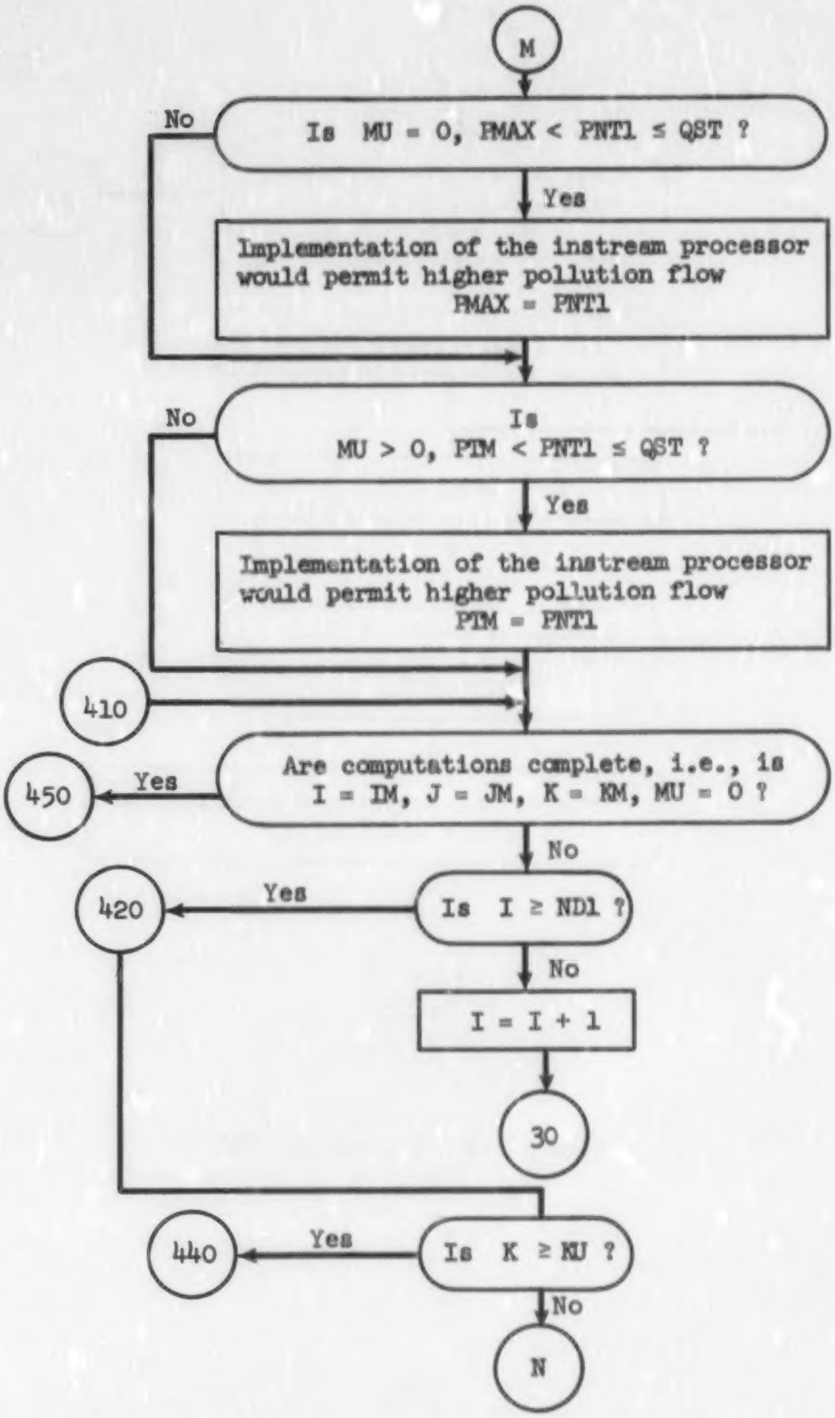

M
No
Is MU = 0, PMAX < PNT1 ≤ QST ?
Yes
Implementation of the instream processor would permit higher pollution flow
PMAX = PNT1
No
Is MU > 0, PTM < PNT1 ≤ QST ?
Yes
Implementation of the instream processor would permit higher pollution flow
PTM = PNT1
410
450
Yes
Are computations complete, i.e., is I = IM, J = JM, K = KM, MU = 0 ?
No
420
Yes
Is I ≥ ND1 ?
No
I = I + 1
30
440
Yes
Is K ≥ NU ?
No
N

Figure C.20 Subroutine PTMX

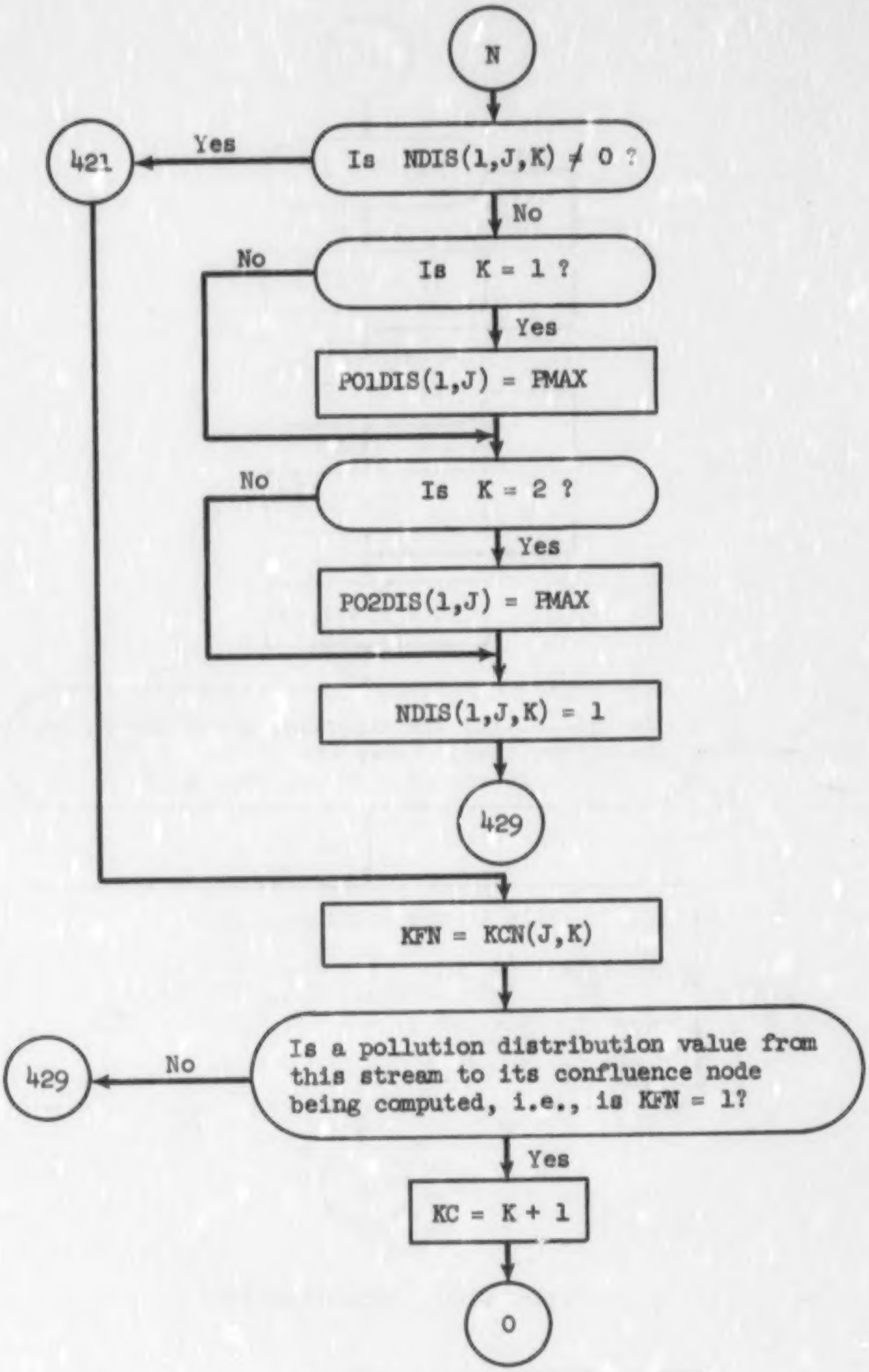

N
Is NDIS(1,J,K) ≠ 0 ?
Yes
421
No
Is K = 1 ?
No
Yes
PO1DIS(1,J) = PMAX
Is K = 2 ?
No
Yes
PO2DIS(1,J) = PMAX
NDIS(1,J,K) = 1
429
KFN = KCN(J,K)
Is a pollution distribution value from this stream to its confluence node being computed, i.e., is KFN = 1?
No
429
Yes
KC = K + 1
O

Figure C.20 Subroutine PTMX

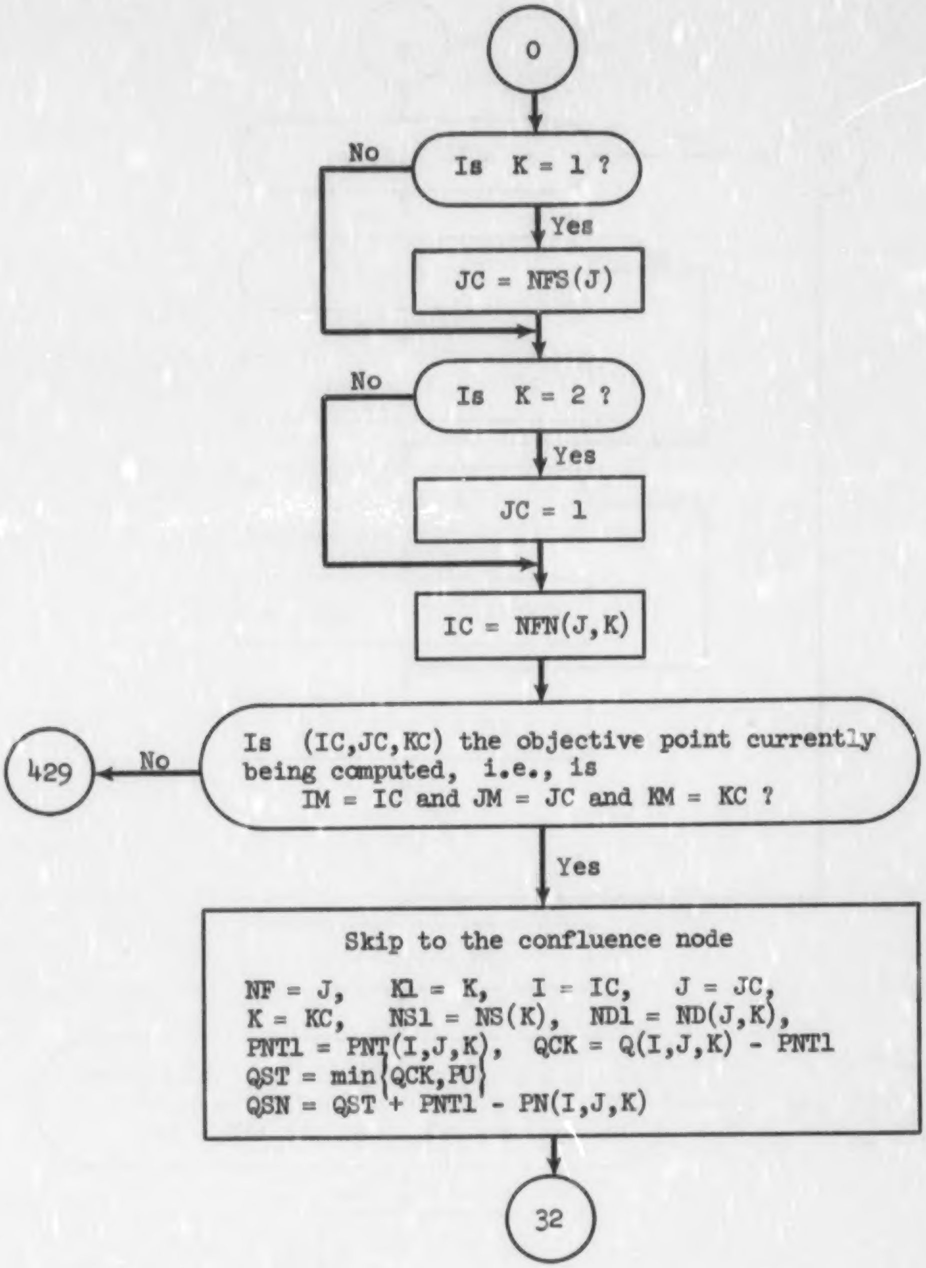

0
Is K = 1 ?
No
Yes
JC = NFS(J)
Is K = 2 ?
No
Yes
JC = 1
IC = NFN(J,K)
Is (IC,JC,KC) the objective point currently being computed, i.e., is IM = IC and JM = JC and KM = KC ?
No
429
Yes
Skip to the confluence node
NF = J, K1 = K, I = IC, J = JC, K = KC, NS1 = NS(K), ND1 = ND(J,K), PNT1 = PNT(I,J,K), QCK = Q(I,J,K) - PNT1
QST = min{QCK,PU}
QSN = QST + PNT1 - PN(I,J,K)
32

Figure C.20 Subroutine PTMX

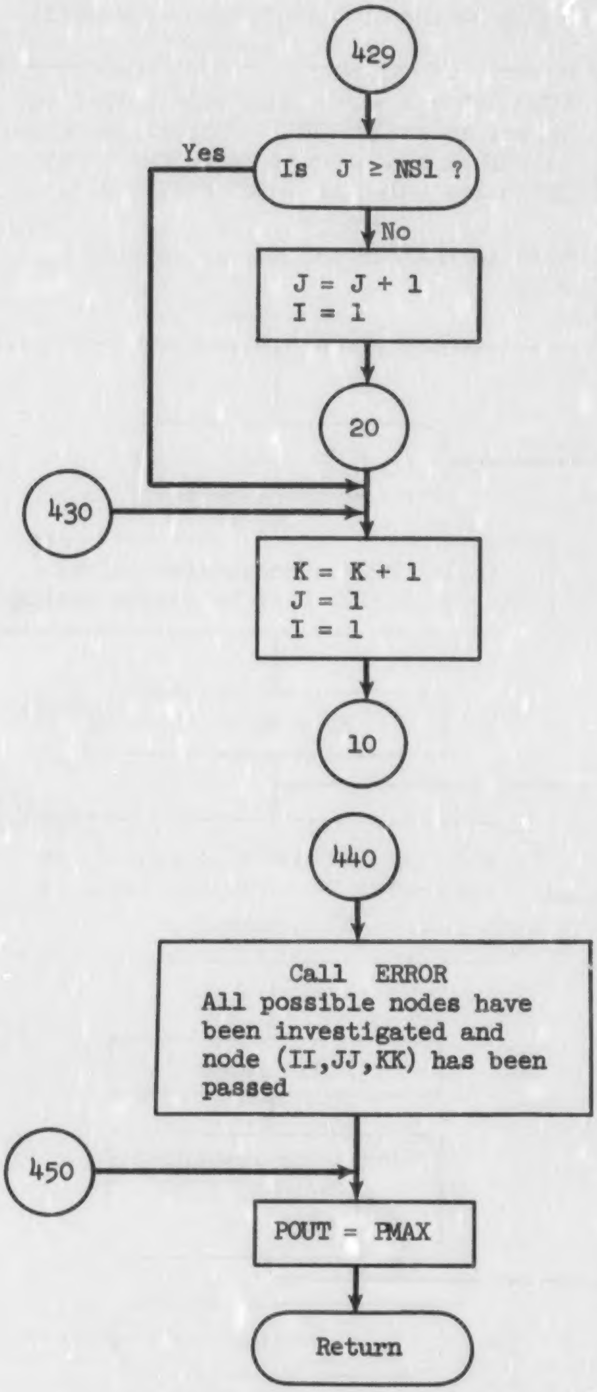

429
Yes
Is J ≥ NS1 ?
No
J = J + 1
I = 1
20
430
K = K + 1
J = 1
I = 1
10
440
Call ERROR
All possible nodes have been investigated and node (II,JJ,KK) has been passed
450
POUT = PMAX
Return

Figure C.20 Subroutine PTMX

Subroutine STORE(PLT,PDIS,MDEM,NDIS)

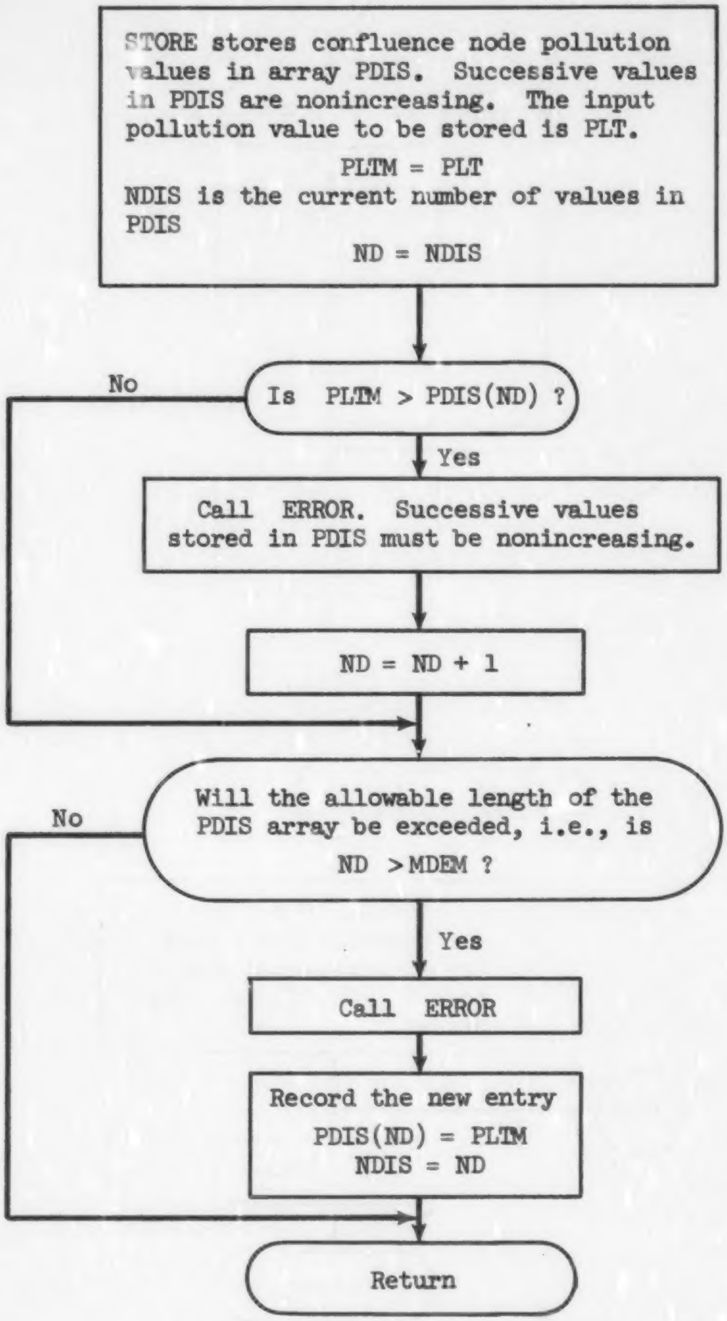
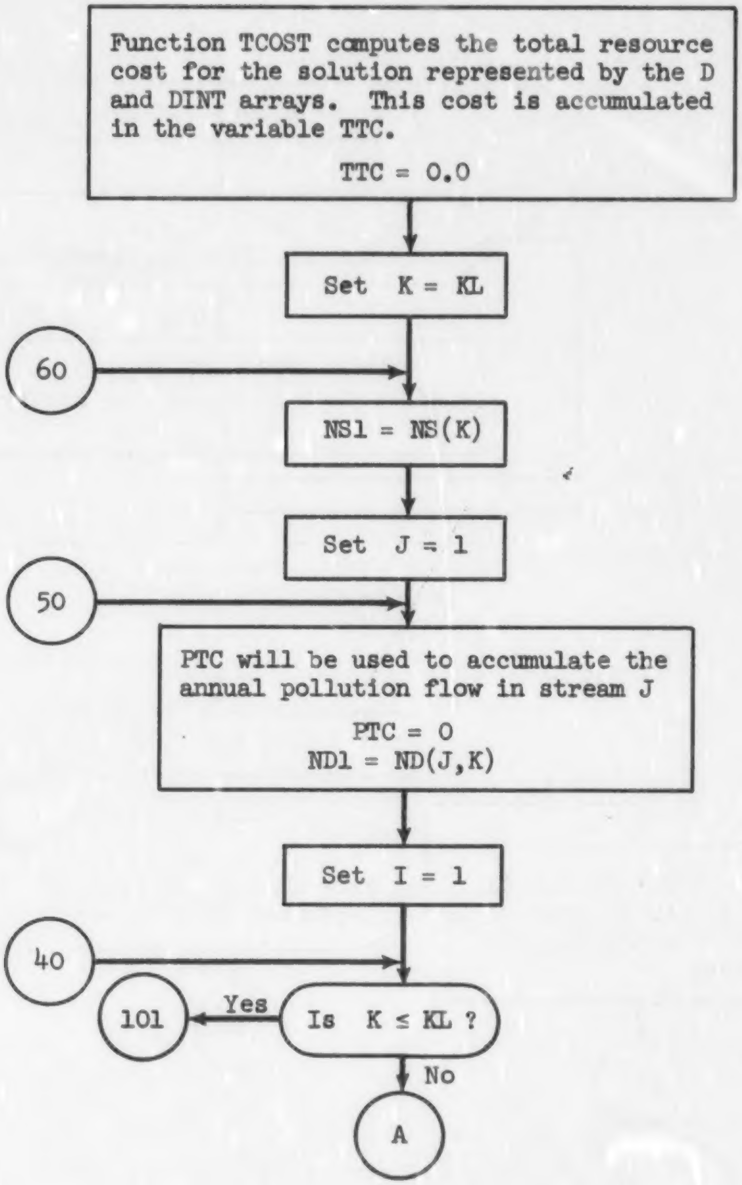

STORE stores confluence node pollution values in array PDIS. Successive values in PDIS are nonincreasing. The input pollution value to be stored is PLT.
PLTM = PLT
NDIS is the current number of values in PDIS
ND = NDIS
Is PLTM > PDIS(ND) ?
No
Yes
Call ERROR. Successive values stored in PDIS must be nonincreasing.
ND = ND + 1
Will the allowable length of the PDIS array be exceeded, i.e., is ND >MDEM ?
No
Yes
Call ERROR
Record the new entry
PDIS(ND) = PLTM
NDIS = ND
Return

Figure C.21 Subroutine STORE

Function TCOST(X)



Function TCOST computes the total resource cost for the solution represented by the D and DINT arrays. This cost is accumulated in the variable TTC.

PTC will be used to accumulate the annual pollution flow in stream J

TTC = 0.0
Set K = KL
60
NS1 = NS(K)
Set J = 1
50
PTC = 0
ND1 = ND(J,K)
Set I = 1
40
101
Yes
Is K ≤ KL ?
No
A

Figure C.22 Function TCOST

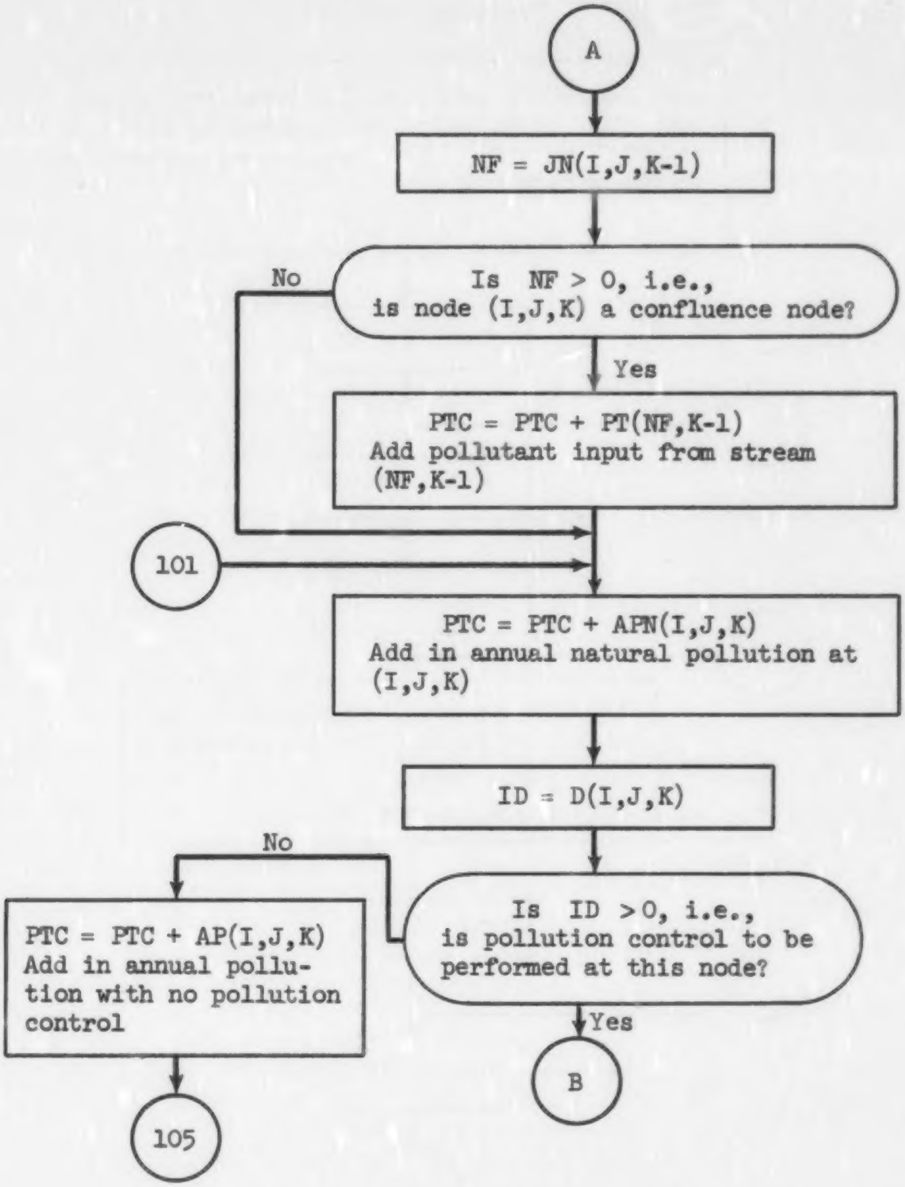

A
NF = JN(I,J,K-1)
No
Is NF > 0, i.e., is node (I,J,K) a confluence node?
Yes
PTC = PTC + PT(NF,K-1) Add pollutant input from stream (NF,K-1)
101
PTC = PTC + APN(I,J,K) Add in annual natural pollution at (I,J,K)
ID = D(I,J,K)
No
Is ID >0, i.e., is pollution control to be performed at this node?
PTC = PTC + AP(I,J,K) Add in annual pollution with no pollution control
Yes
B
105

Figure C.22 Function TCOST

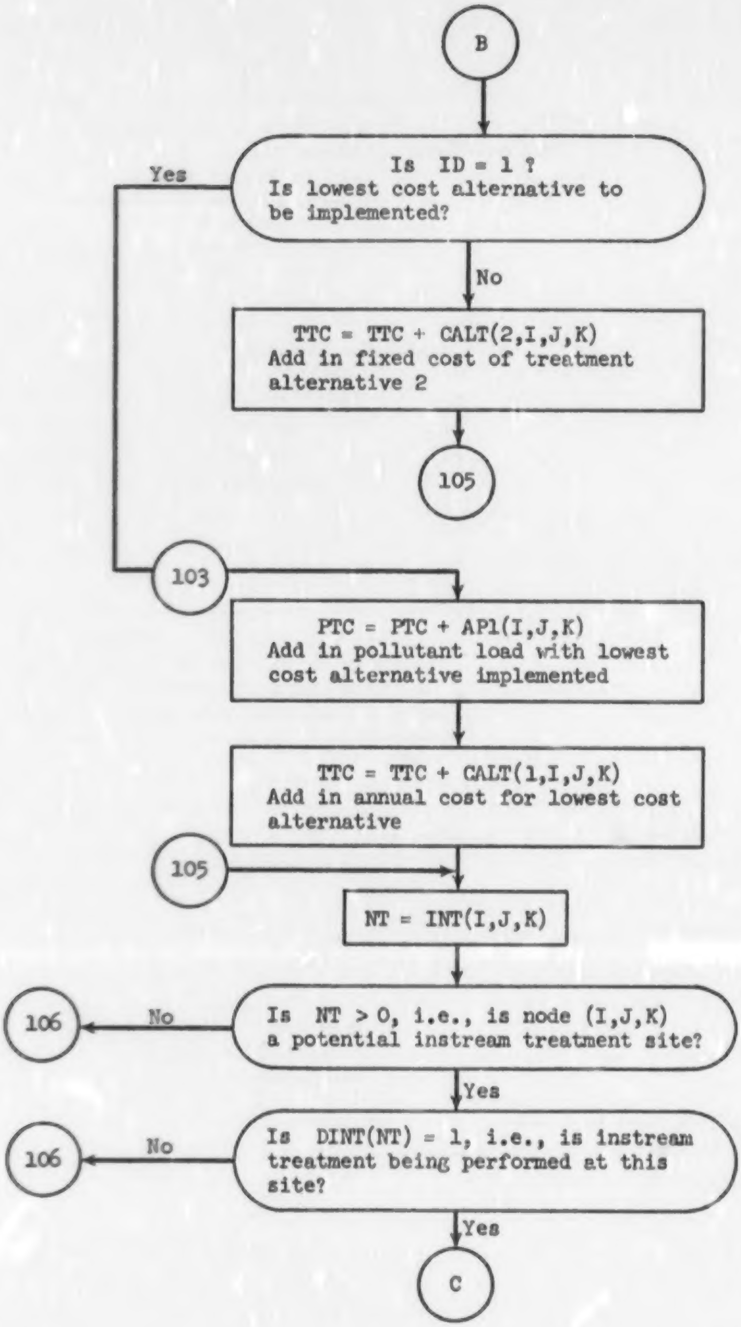

B
Is ID = 1 ?
Is lowest cost alternative to be implemented?
Yes
No
TTC = TTC + CALT(2,I,J,K)
Add in fixed cost of treatment alternative 2
105
103
PTC = PTC + AP1(I,J,K)
Add in pollutant load with lowest cost alternative implemented
TTC = TTC + CALT(1,I,J,K)
Add in annual cost for lowest cost alternative
105
NT = INT(I,J,K)
Is NT > 0, i.e., is node (I,J,K) a potential instream treatment site?
No
106
Yes
Is DINT(NT) = 1, i.e., is instream treatment being performed at this site?
No
106
Yes
C

Figure C.22 Function TCOST

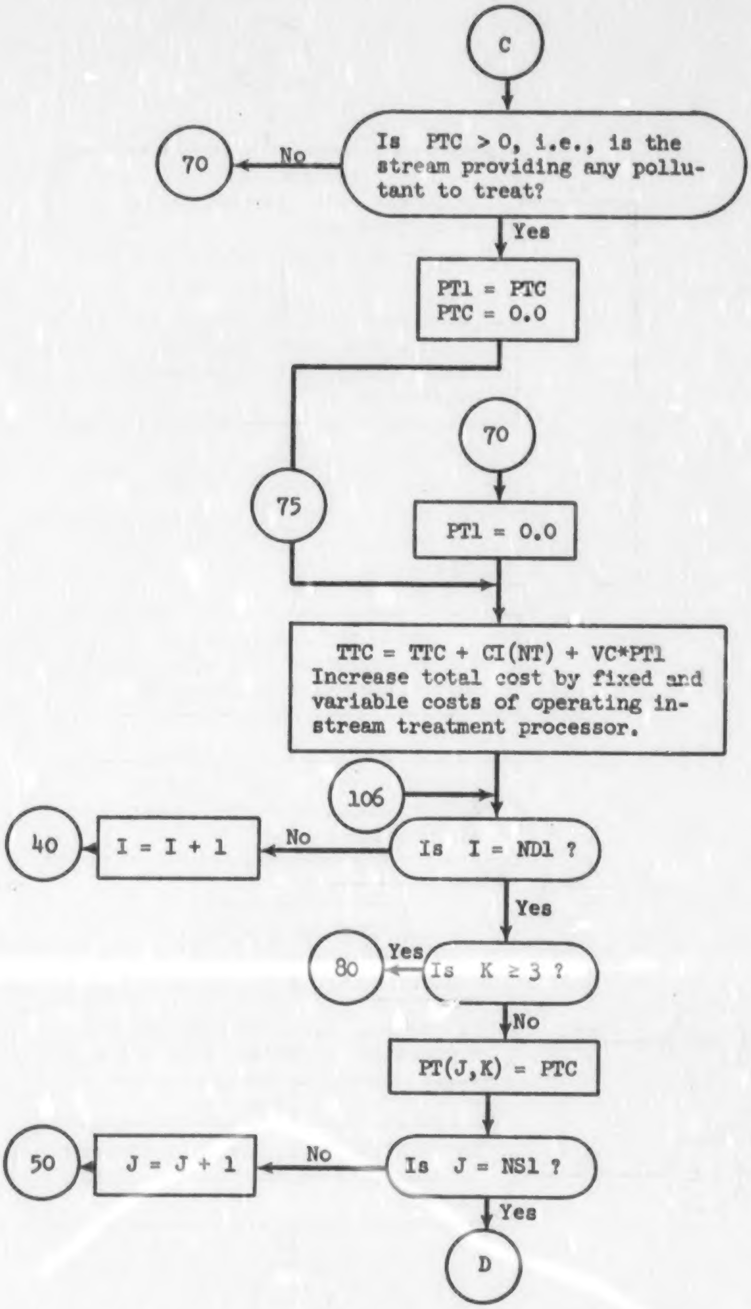

C
Is PTC > 0, i.e., is the stream providing any pollutant to treat?
No
70
Yes
PT1 = PTC
PTC = 0.0
70
75
PT1 = 0.0
TTC = TTC + CI(NT) + VC*PT1
Increase total cost by fixed and variable costs of operating instream treatment processor.
106
Is I = ND1 ?
No
I = I + 1
40
Yes
Is K ≥ 3 ?
Yes
80
No
PT(J,K) = PTC
Is J = NS1 ?
No
J = J + 1
50
Yes
D

Figure C.22 Function TCOST

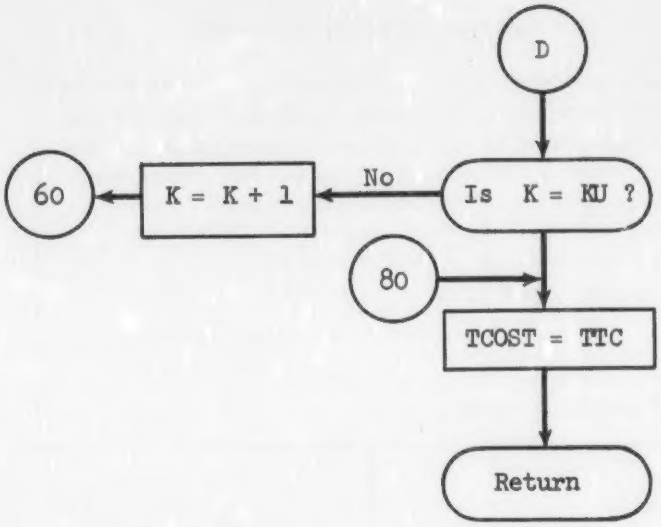
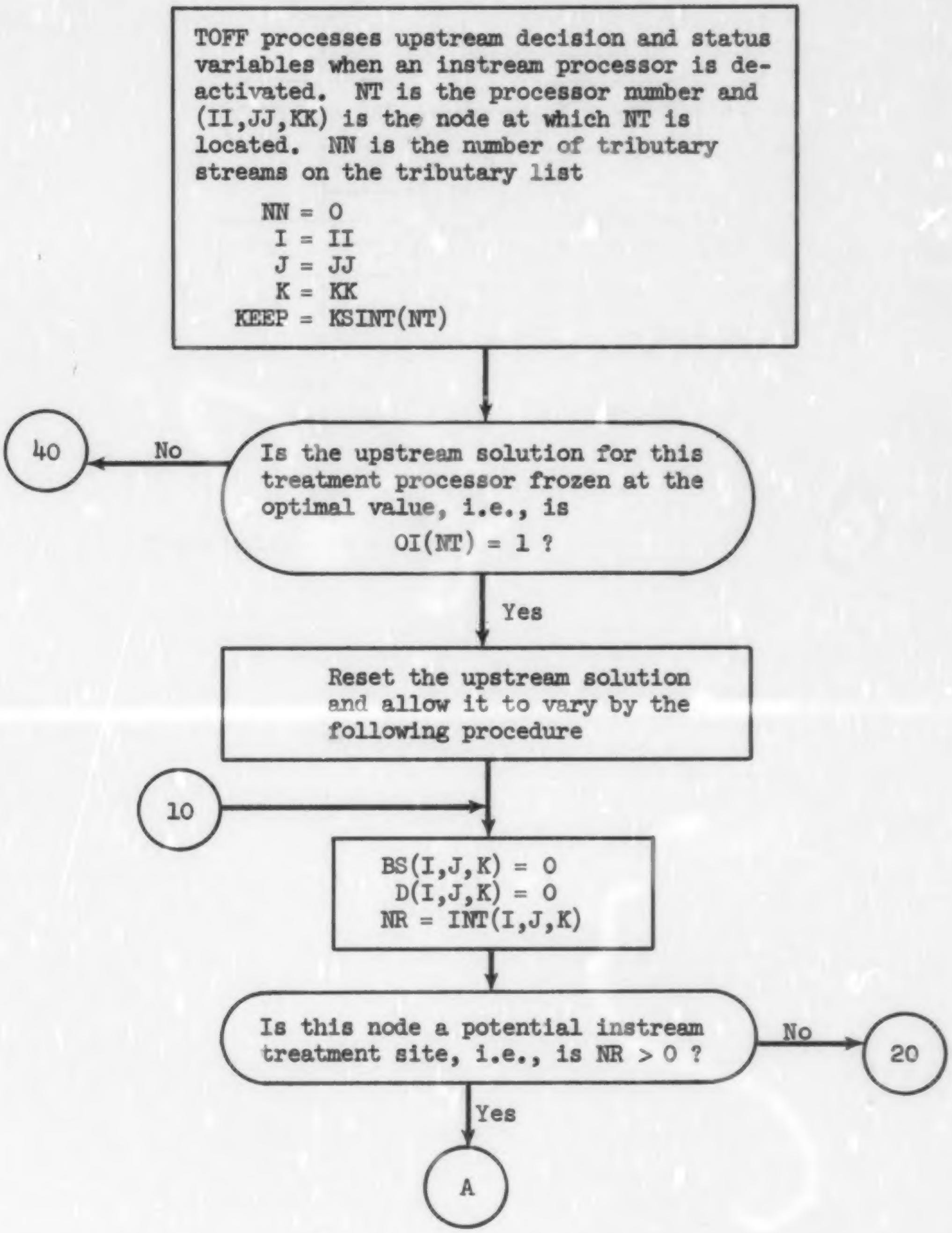

D
No
Is K = KU ?
K = K + 1
60
80
TCOST = TTC
Return

Figure C.22 Function TCOST

Subroutine TOFF(NT,II,JJ,KK)



TOFF processes upstream decision and status variables when an instream processor is deactivated. NT is the processor number and (II,JJ,KK) is the node at which NT is located. NN is the number of tributary streams on the tributary list

NN = 0
I = II
J = JJ
K = KK
KEEP = KSINT(NT)

Is the upstream solution for this treatment processor frozen at the optimal value, i.e., is $OI(NT) = 1$?

Reset the upstream solution and allow it to vary by the following procedure

BS(I,J,K) = 0
D(I,J,K) = 0
NR = INT(I,J,K)

Is this node a potential instream treatment site, i.e., is $NR > 0$?

Figure C.23 Subroutine TOFF

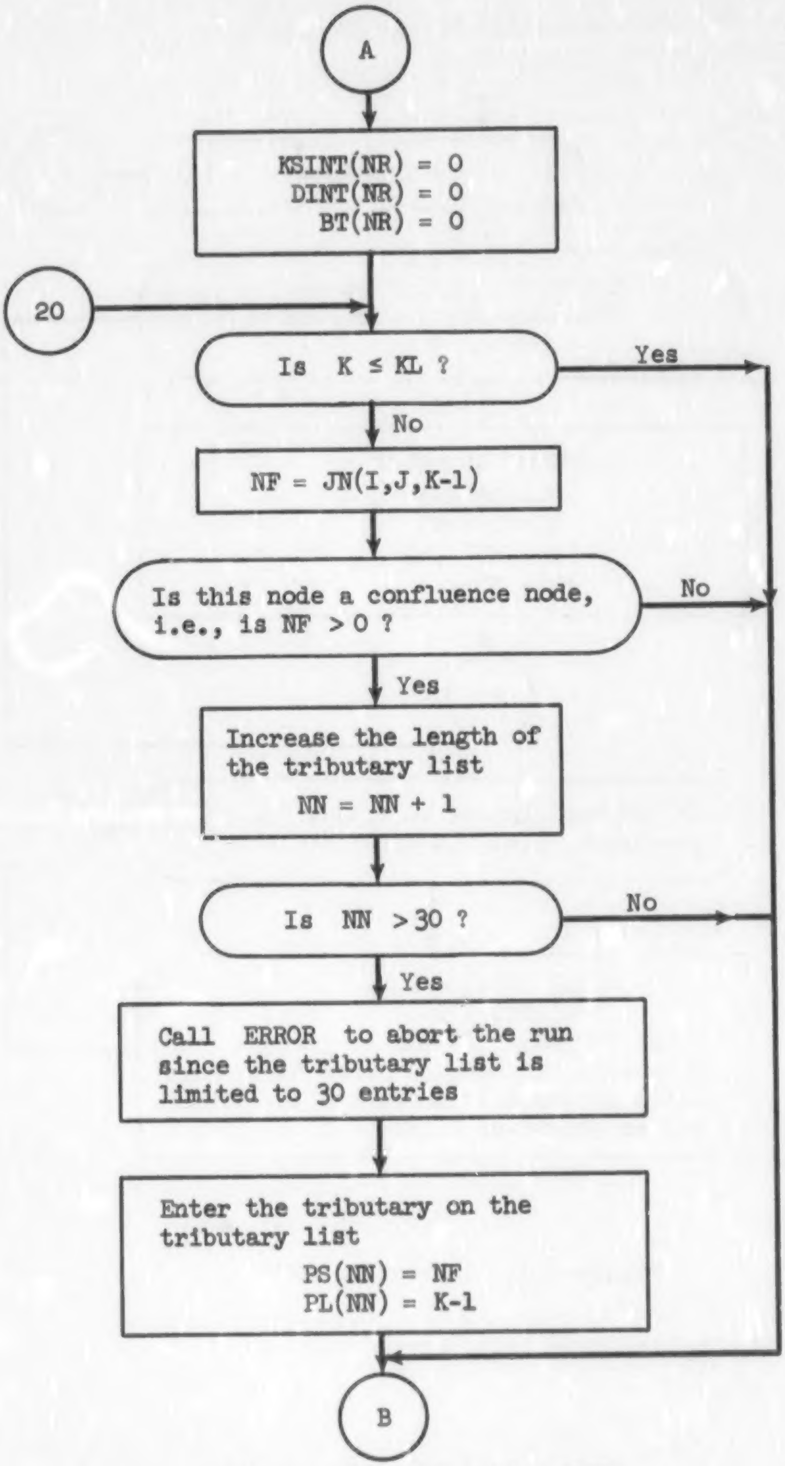

A
KSINT(NR) = 0
DINT(NR) = 0
BT(NR) = 0
20
Is K ≤ KL ?
Yes
No
NF = JN(I,J,K-1)
Is this node a confluence node, i.e., is NF > 0 ?
No
Yes
Increase the length of the tributary list
NN = NN + 1
Is NN > 30 ?
No
Yes
Call ERROR to abort the run since the tributary list is limited to 30 entries
Enter the tributary on the tributary list
PS(NN) = NF
PL(NN) = K-1
B

Figure C.23 Subroutine TOFF

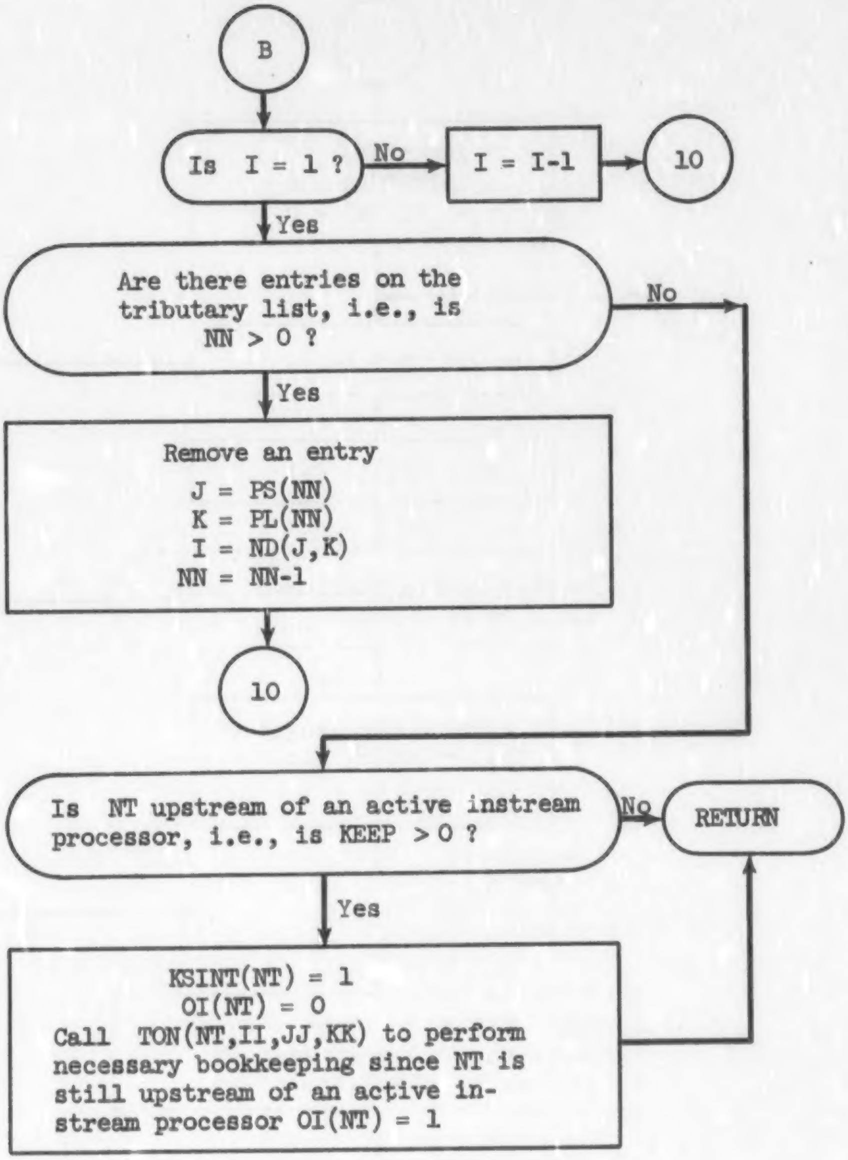

B
Is I = 1 ?
No
I = I-1
10
Yes
Are there entries on the tributary list, i.e., is NN > 0 ?
No
Yes
Remove an entry
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN-1
10
Is NT upstream of an active instream processor, i.e., is KEEP > 0 ?
No
RETURN
Yes
KSINT(NT) = 1
OI(NT) = 0
Call TON(NT,II,JJ,KK) to perform necessary bookkeeping since NT is still upstream of an active in-stream processor OI(NT) = 1

Figure C.23 Subroutine TOFF

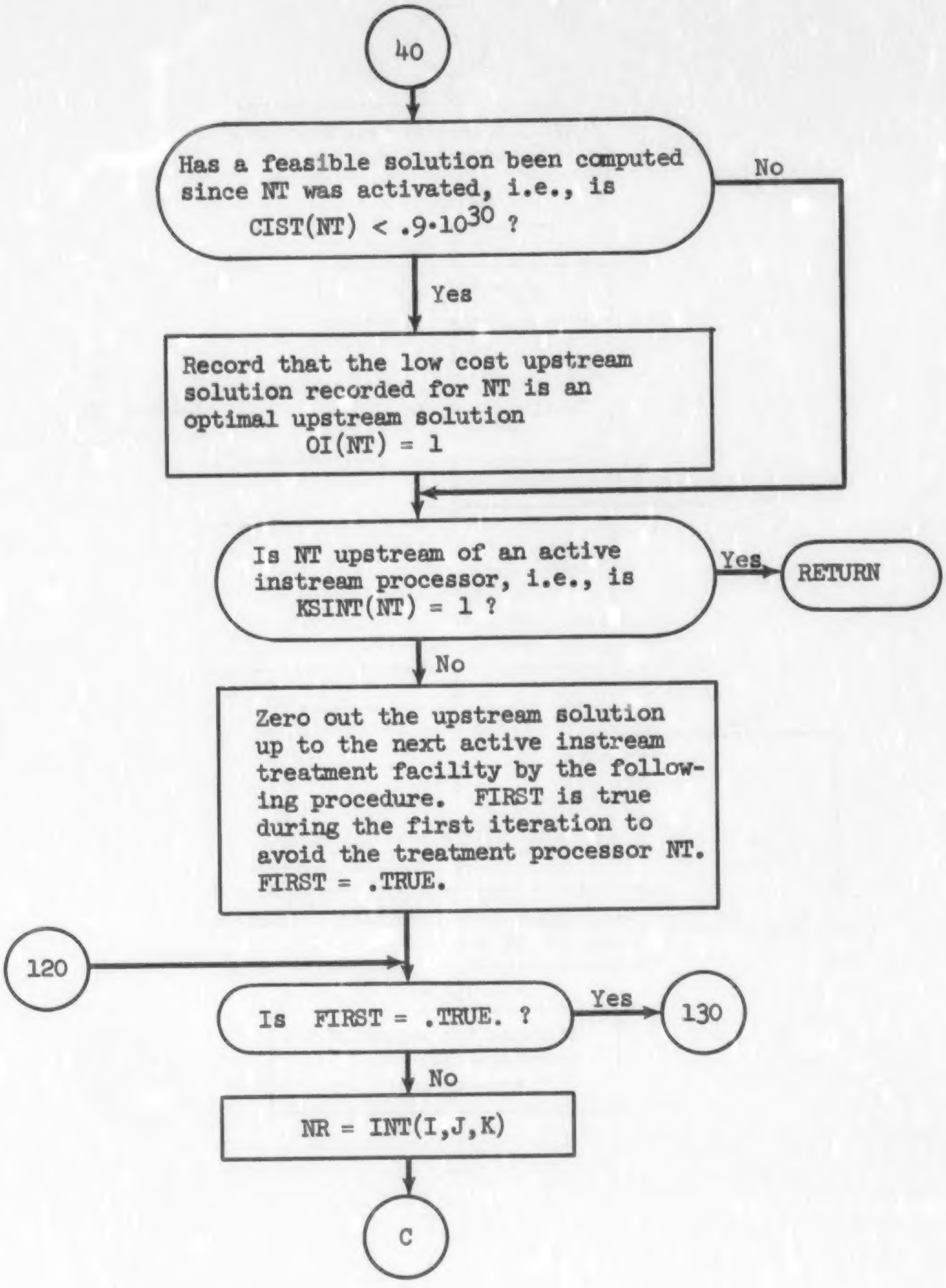

40
Has a feasible solution been computed since NT was activated, i.e., is CIST(NT) < .9·10^30 ?
No
Yes
Record that the low cost upstream solution recorded for NT is an optimal upstream solution OI(NT) = 1
Is NT upstream of an active instream processor, i.e., is KSINT(NT) = 1 ?
Yes
RETURN
No
Zero out the upstream solution up to the next active instream treatment facility by the following procedure. FIRST is true during the first iteration to avoid the treatment processor NT. FIRST = .TRUE.
120
Is FIRST = .TRUE. ?
Yes
130
No
NR = INT(I,J,K)
C

Figure C.23 Subroutine TOFF

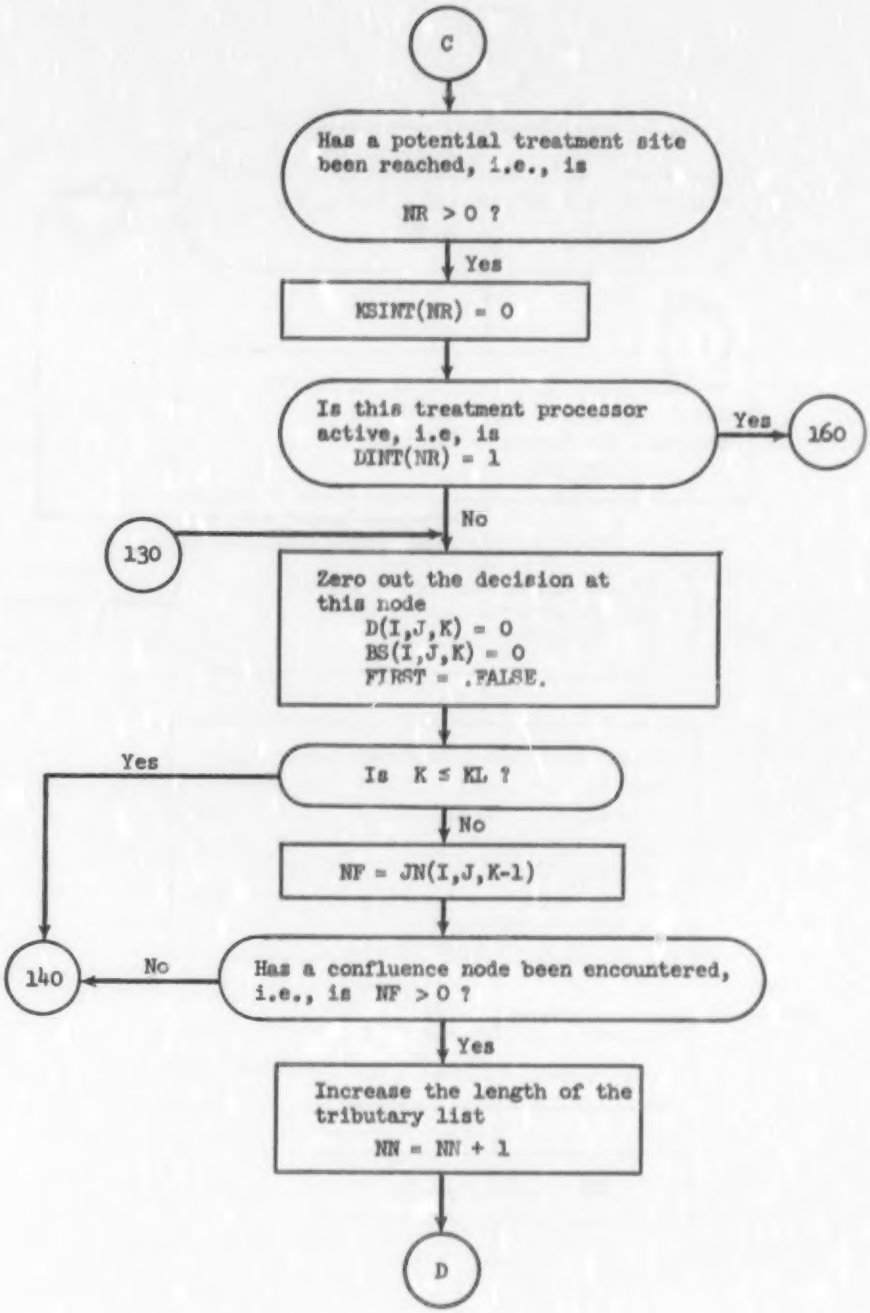

C
Has a potential treatment site been reached, i.e., is NR > 0 ?
Yes
KSINT(NR) = 0
Is this treatment processor active, i.e, is DINT(NR) = 1
Yes
160
No
130
Zero out the decision at this node
D(I,J,K) = 0
BS(I,J,K) = 0
FIRST = .FALSE.
Yes
Is K ≤ KL ?
No
NF = JN(I,J,K-1)
140
No
Has a confluence node been encountered, i.e., is NF > 0 ?
Yes
Increase the length of the tributary list
NN = NN + 1
D

Figure C.23 Subroutine TOFF

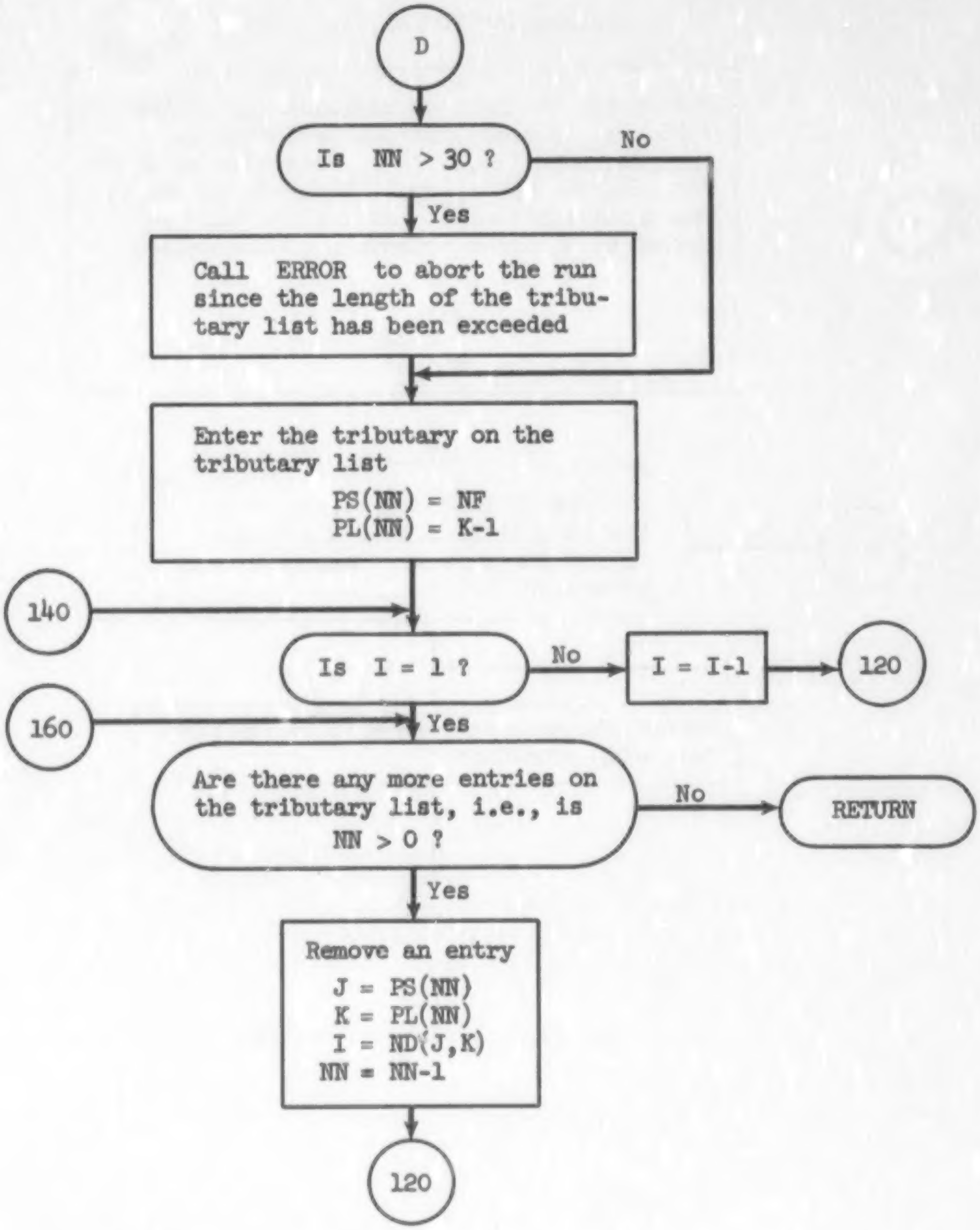

D
Is NN > 30 ?
No
Yes
Call ERROR to abort the run since the length of the tributary list has been exceeded
Enter the tributary on the tributary list
PS(NN) = NF
PL(NN) = K-1
140
Is I = 1 ?
No
I = I-1
120
160
Yes
Are there any more entries on the tributary list, i.e., is NN > 0 ?
No
RETURN
Yes
Remove an entry
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN-1
120

Figure C.23 Subroutine TOFF

Subroutine TON(NT,II,JJ,KK)

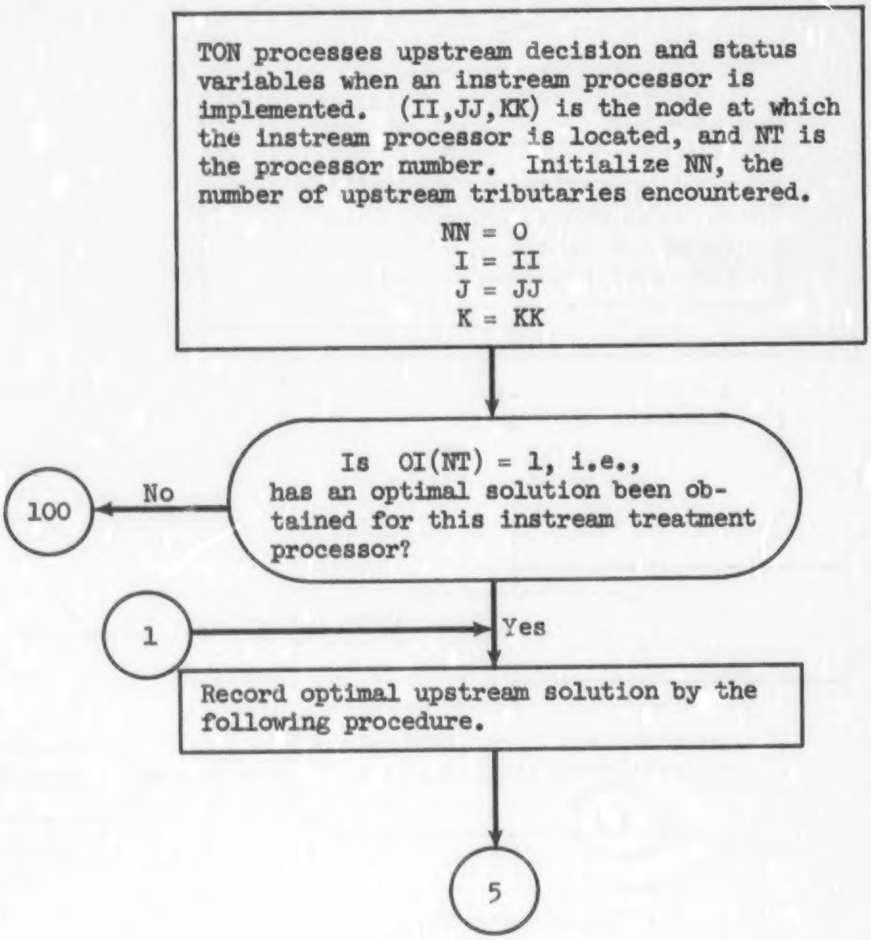

TON processes upstream decision and status variables when an instream processor is implemented. (II,JJ,KK) is the node at which the instream processor is located, and NT is the processor number. Initialize NN, the number of upstream tributaries encountered.
NN = 0
I = II
J = JJ
K = KK
Is OI(NT) = 1, i.e., has an optimal solution been obtained for this instream treatment processor?
No
100
Yes
1
Record optimal upstream solution by the following procedure.
5

Figure C.24 Subroutine TON

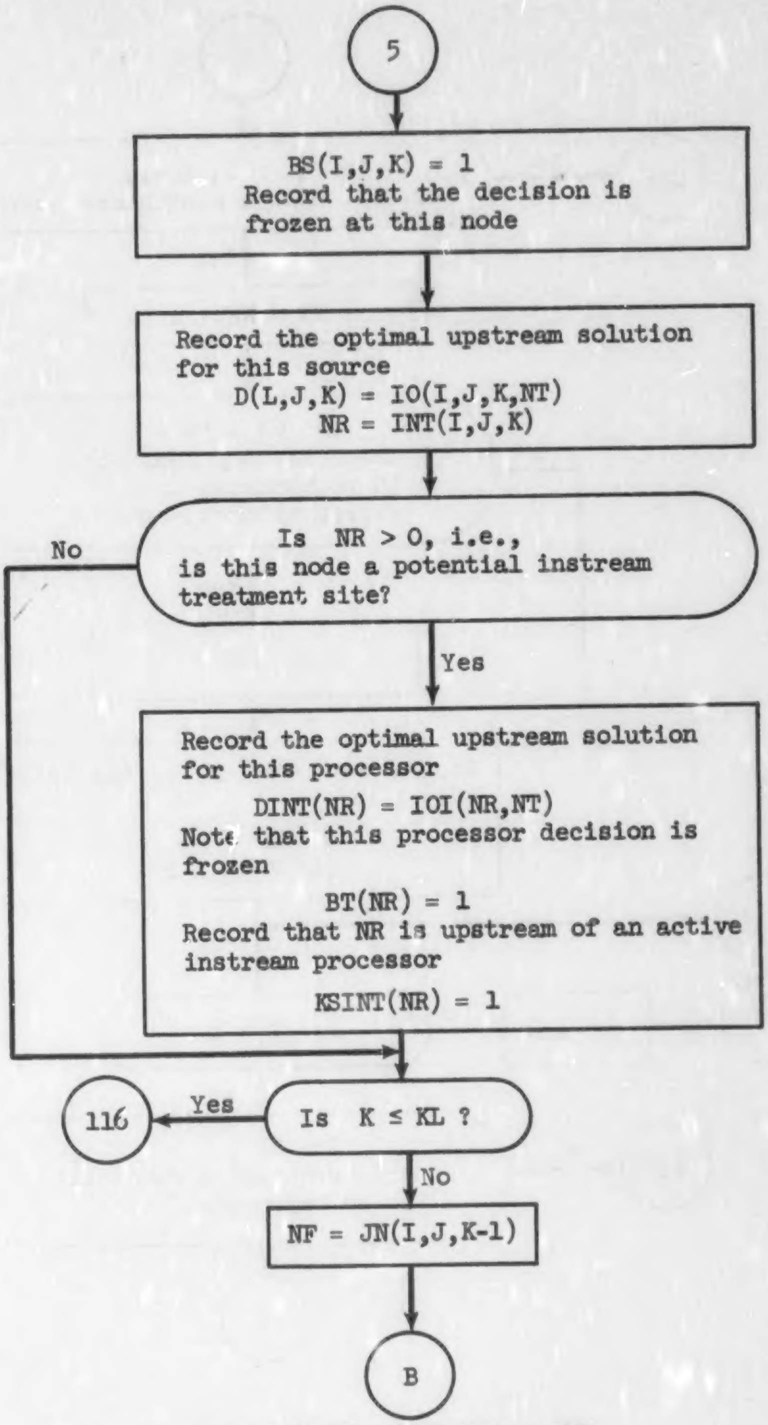

5
BS(I,J,K) = 1
Record that the decision is frozen at this node
Record the optimal upstream solution for this source
D(L,J,K) = IO(I,J,K,NT)
NR = INT(I,J,K)
Is NR > 0, i.e., is this node a potential instream treatment site?
No
Yes
Record the optimal upstream solution for this processor
DINT(NR) = IOI(NR,NT)
Note that this processor decision is frozen
BT(NR) = 1
Record that NR is upstream of an active instream processor
KSINT(NR) = 1
116
Yes
Is K ≤ KL ?
No
NF = JN(I,J,K-1)
B

Figure C.24 Subroutine TON

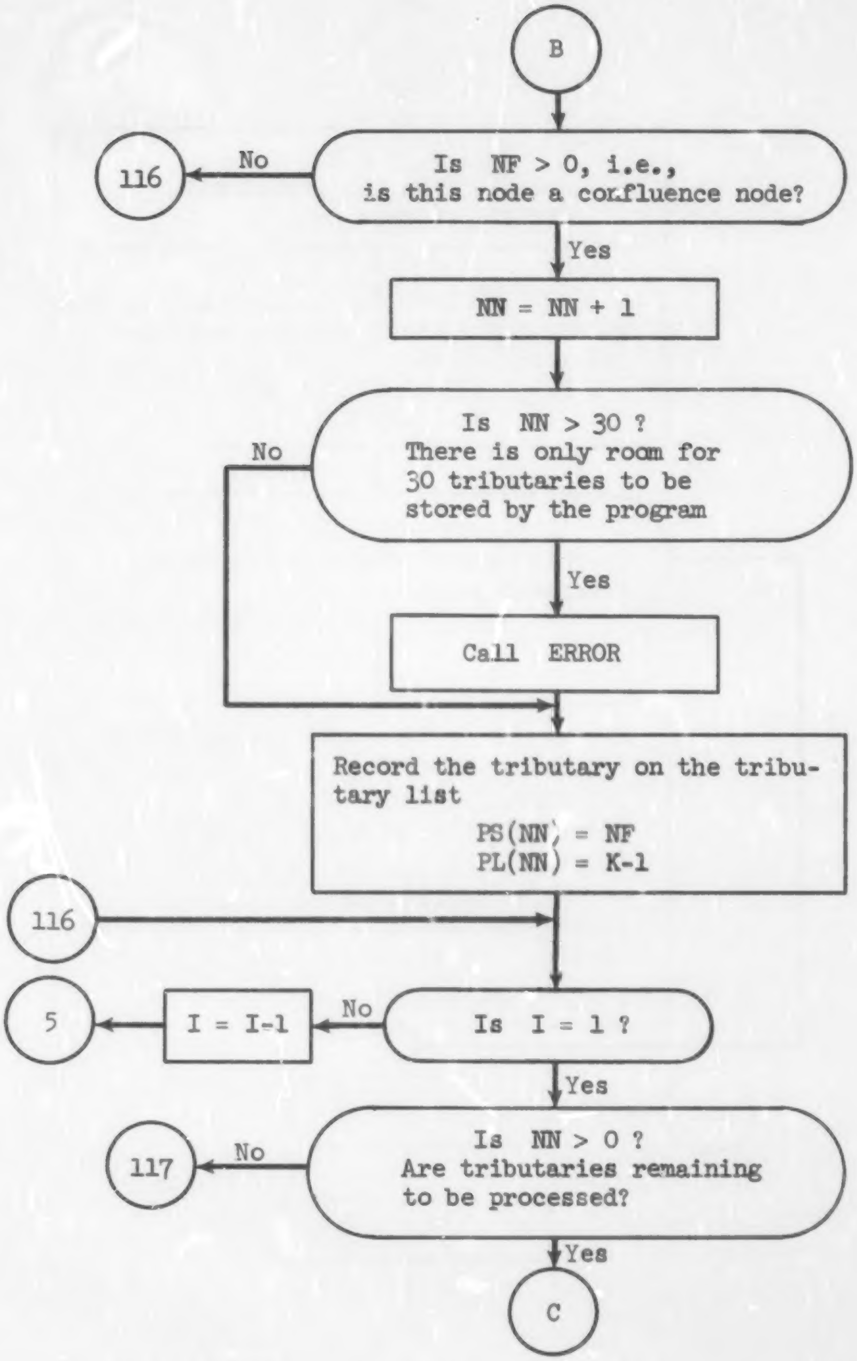
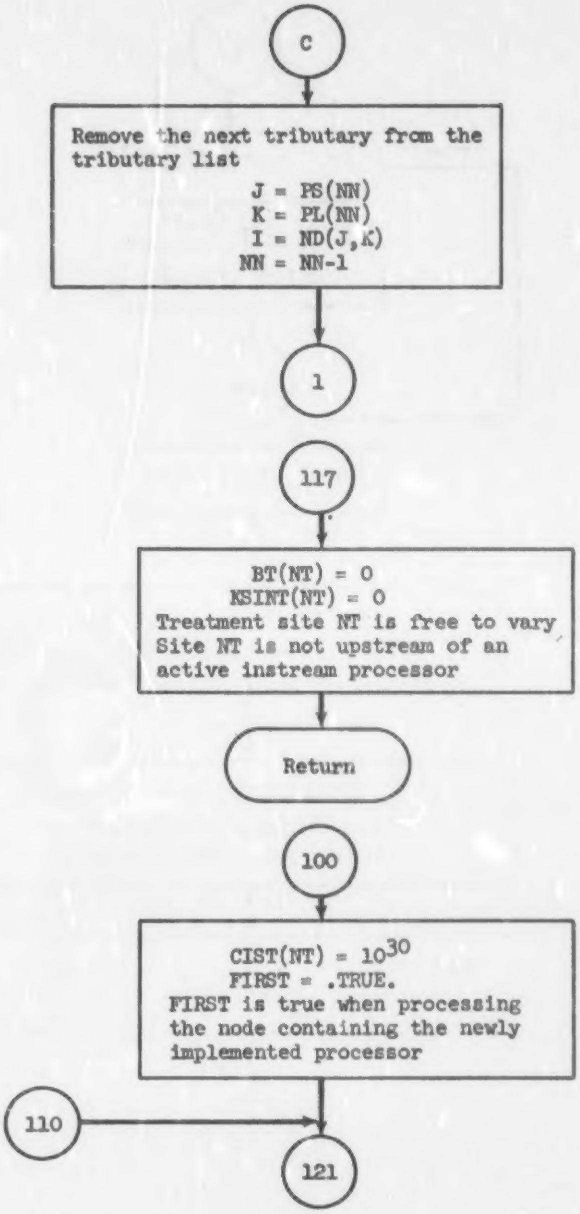

B
Is NF > 0, i.e.,
is this node a confluence node?
No
116
Yes
NN = NN + 1
Is NN > 30 ?
There is only room for
30 tributaries to be
stored by the program
No
Yes
Call ERROR
Record the tributary on the tributary list
PS(NN) = NF
PL(NN) = K-1
116
Is I = 1 ?
No
I = I-1
5
Yes
Is NN > 0 ?
Are tributaries remaining
to be processed?
No
117
Yes
C

Figure C.24 Subroutine TON



Remove the next tributary from the tributary list

```
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN-1
```

```
BT(NT) = 0
KSINT(NT) = 0
```

Treatment site NT is free to vary
Site NT is not upstream of an active instream processor

$CIST(NT) = 10^{30}$

```
FIRST = .TRUE.
```

FIRST is true when processing the node containing the newly implemented processor

Figure C.24 Subroutine TON

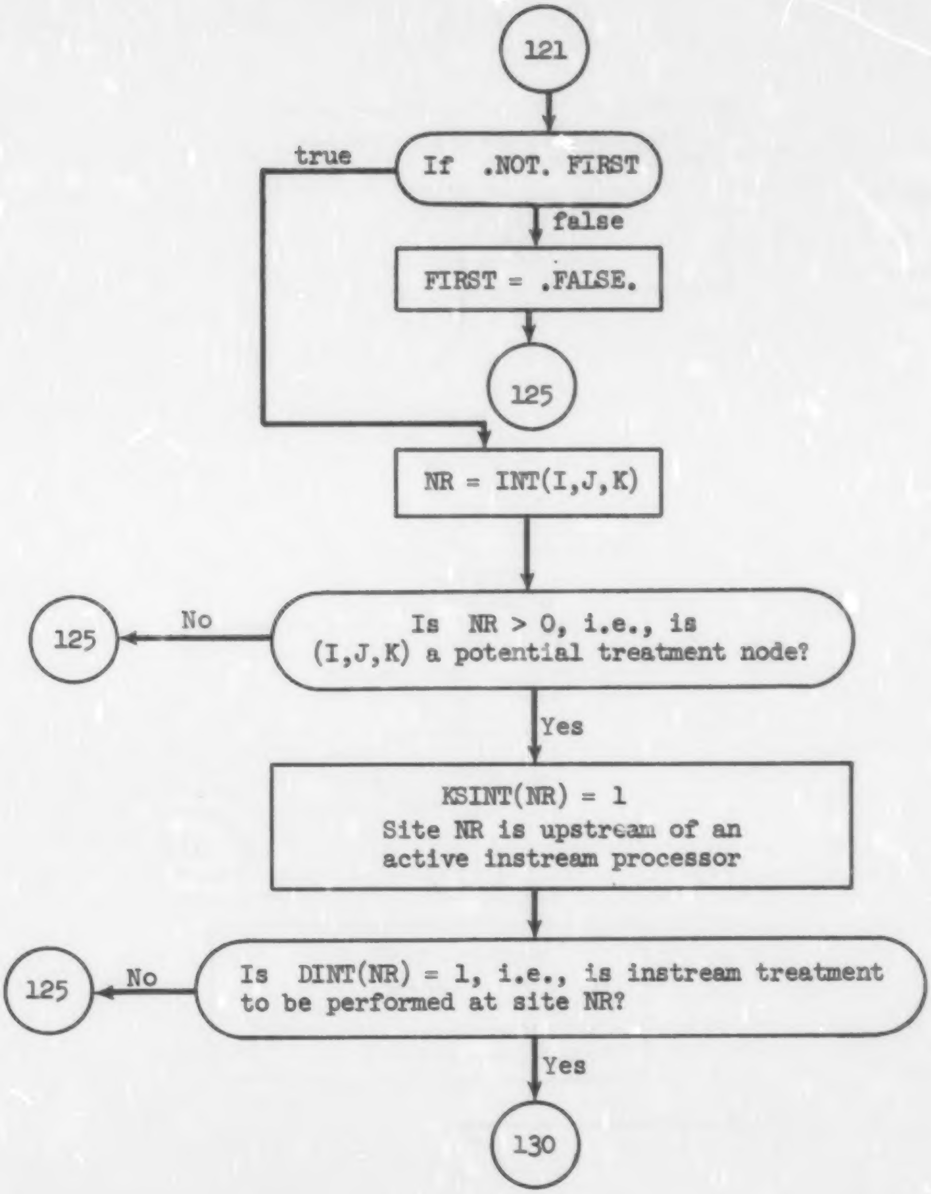

121
true
If .NOT. FIRST
false
FIRST = .FALSE.
125
NR = INT(I,J,K)
125
No
Is NR > 0, i.e., is (I,J,K) a potential treatment node?
Yes
KSINT(NR) = 1
Site NR is upstream of an active instream processor
125
No
Is DINT(NR) = 1, i.e., is instream treatment to be performed at site NR?
Yes
130

Figure C.24 Subroutine TON

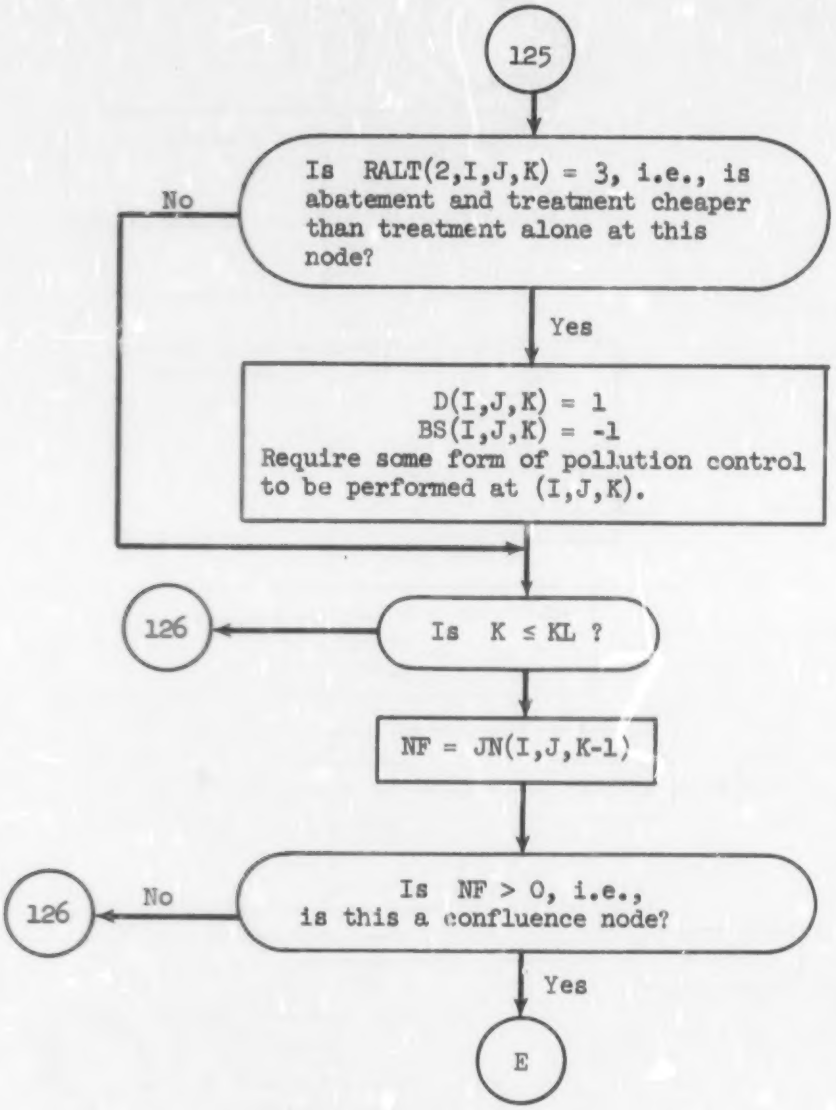

125
Is RALT(2,I,J,K) = 3, i.e., is abatement and treatment cheaper than treatment alone at this node?
No
Yes
D(I,J,K) = 1
BS(I,J,K) = -1
Require some form of pollution control to be performed at (I,J,K).
126
Is K ≤ KL ?
NF = JN(I,J,K-1)
126
No
Is NF > 0, i.e., is this a confluence node?
Yes
E

Figure C.24 Subroutine TON

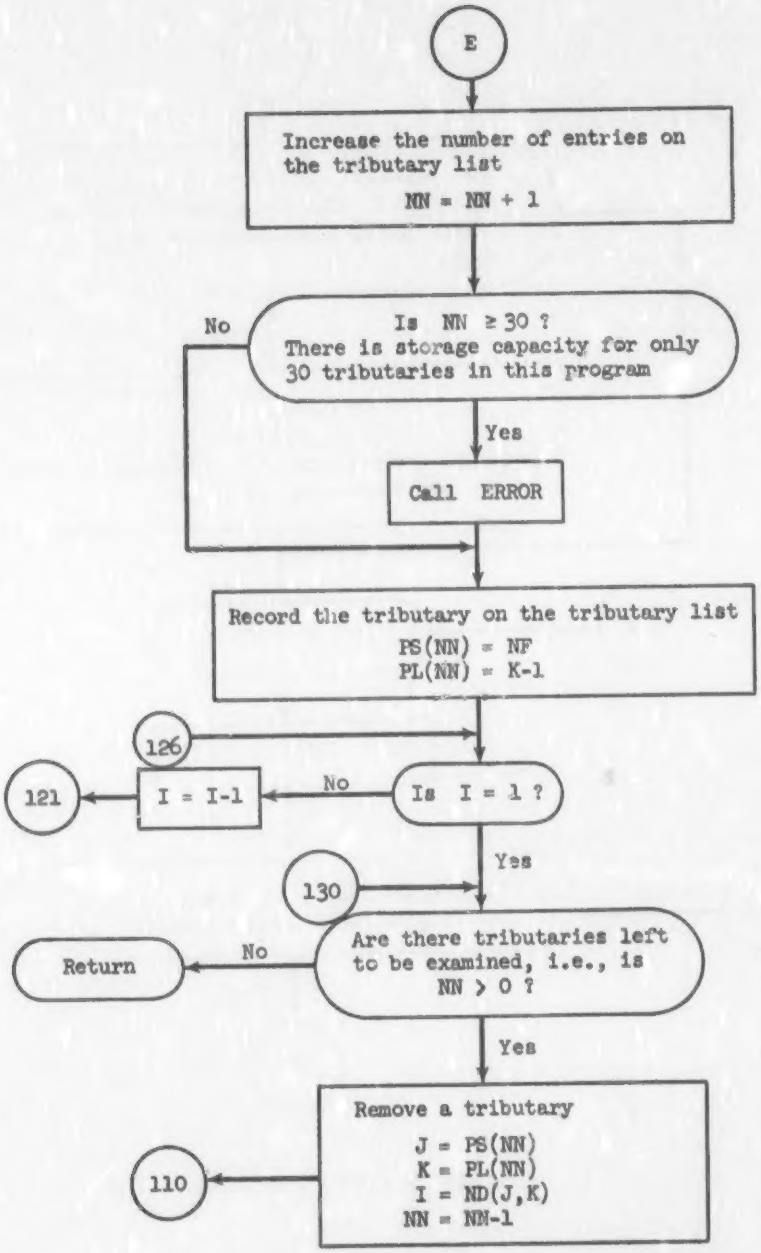

E
Increase the number of entries on the tributary list
NN = NN + 1
Is NN ≥ 30 ?
There is storage capacity for only 30 tributaries in this program
No
Yes
Call ERROR
Record the tributary on the tributary list
PS(NN) = NF
PL(NN) = K-1
126
Is I = 1 ?
No
I = I-1
121
Yes
130
Are there tributaries left to be examined, i.e., is NN > 0 ?
No
Return
Yes
Remove a tributary
J = PS(NN)
K = PL(NN)
I = ND(J,K)
NN = NN-1
110

Figure C.24 Subroutine TON

Subroutine ZO(I,J,K)

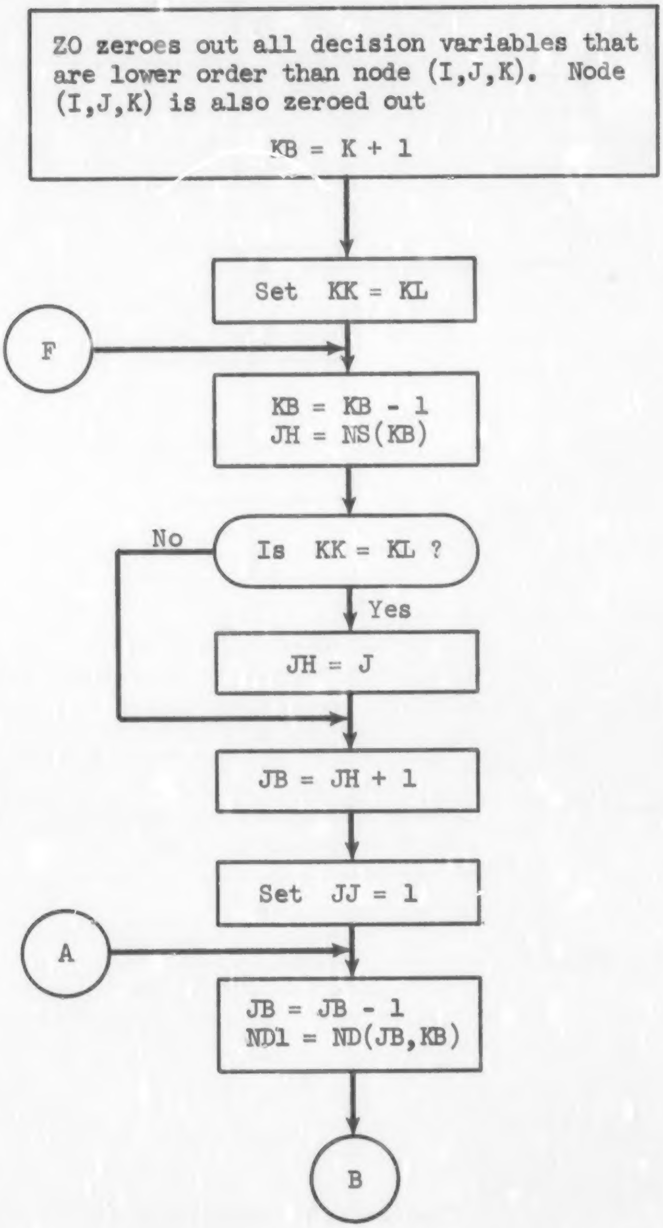

ZO zeroes out all decision variables that are lower order than node (I,J,K). Node (I,J,K) is also zeroed out
KB = K + 1
Set KK = KL
F
KB = KB - 1
JH = NS(KB)
No
Is KK = KL ?
Yes
JH = J
JB = JH + 1
Set JJ = 1
A
JB = JB - 1
ND1 = ND(JB,KB)
B

Figure C.25 Subroutine ZO

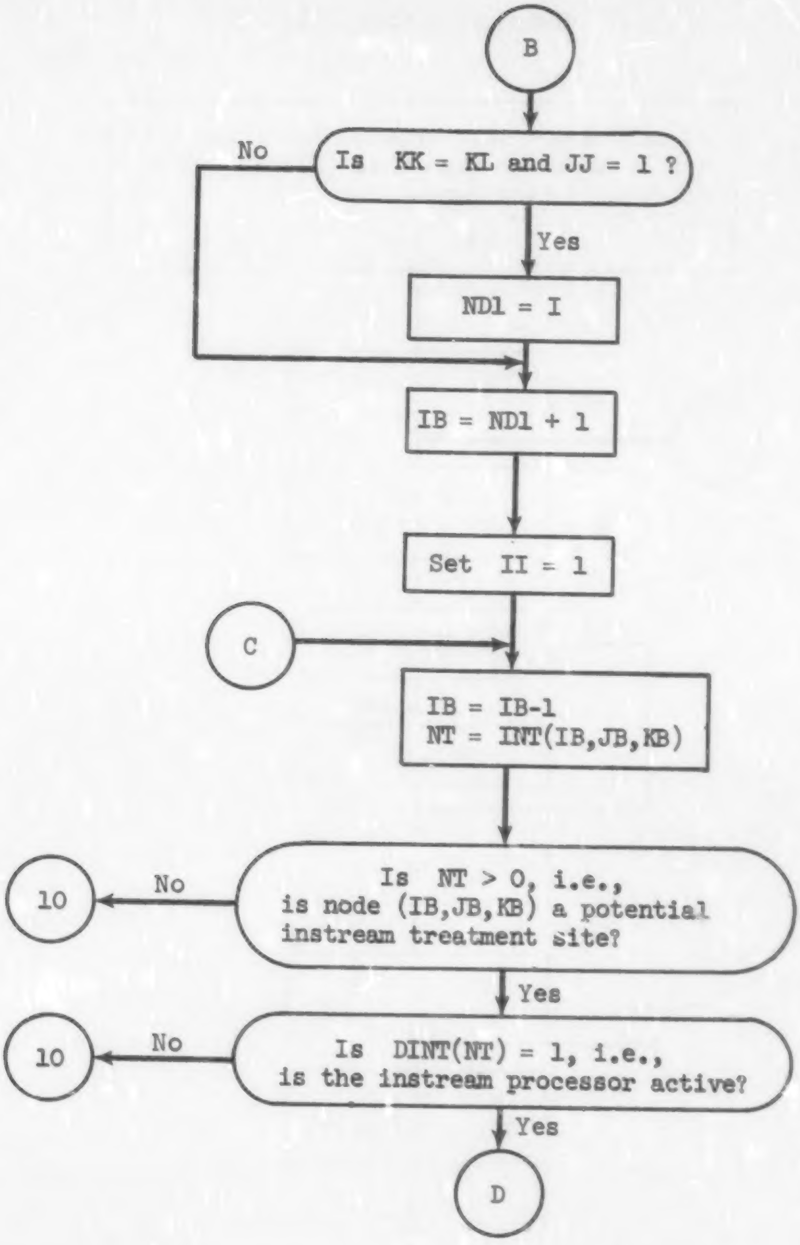

B
Is KK = KL and JJ = 1 ?
No
Yes
ND1 = I
IB = ND1 + 1
Set II = 1
C
IB = IB-1
NT = INT(IB,JB,KB)
Is NT > 0, i.e., is node (IB,JB,KB) a potential instream treatment site?
No
10
Yes
Is DINT(NT) = 1, i.e., is the instream processor active?
No
10
Yes
D

Figure C.25 Subroutine ZO

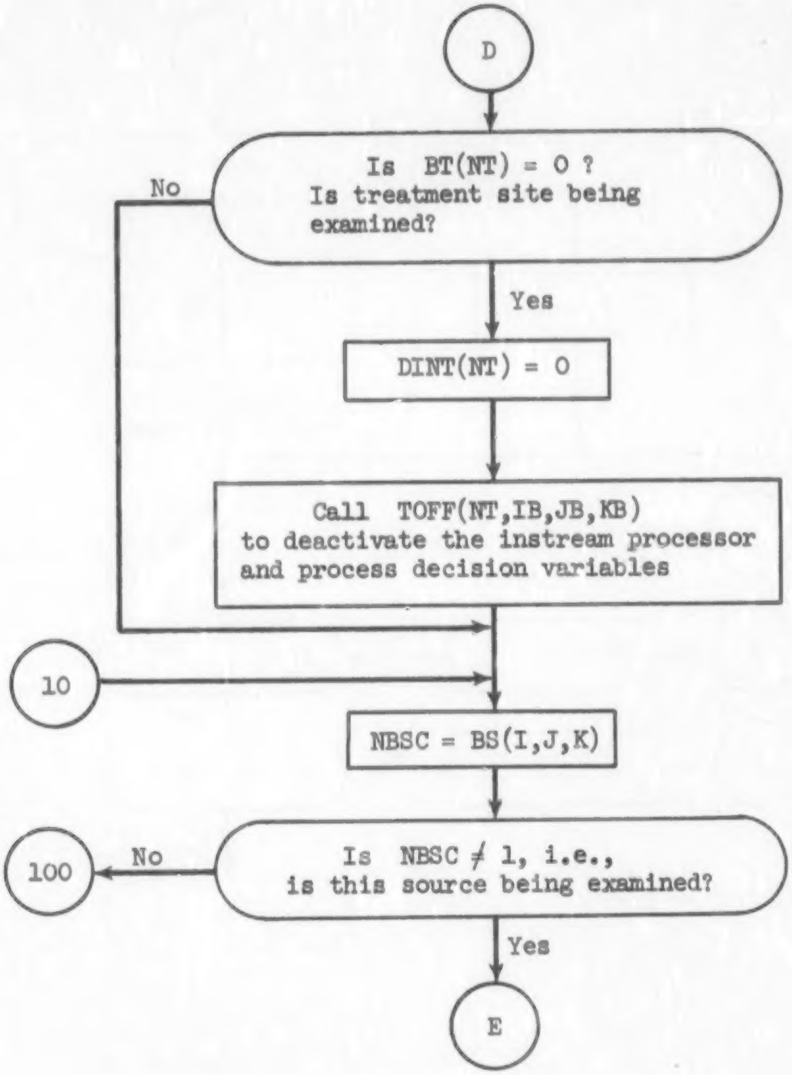

D
Is BT(NT) = 0 ?
Is treatment site being examined?
No
Yes
DINT(NT) = 0
Call TOFF(NT,IB,JB,KB)
to deactivate the instream processor and process decision variables
10
NBSC = BS(I,J,K)
Is NBSC ≠ 1, i.e., is this source being examined?
No
100
Yes
E

Figure C.25 Subroutine ZO

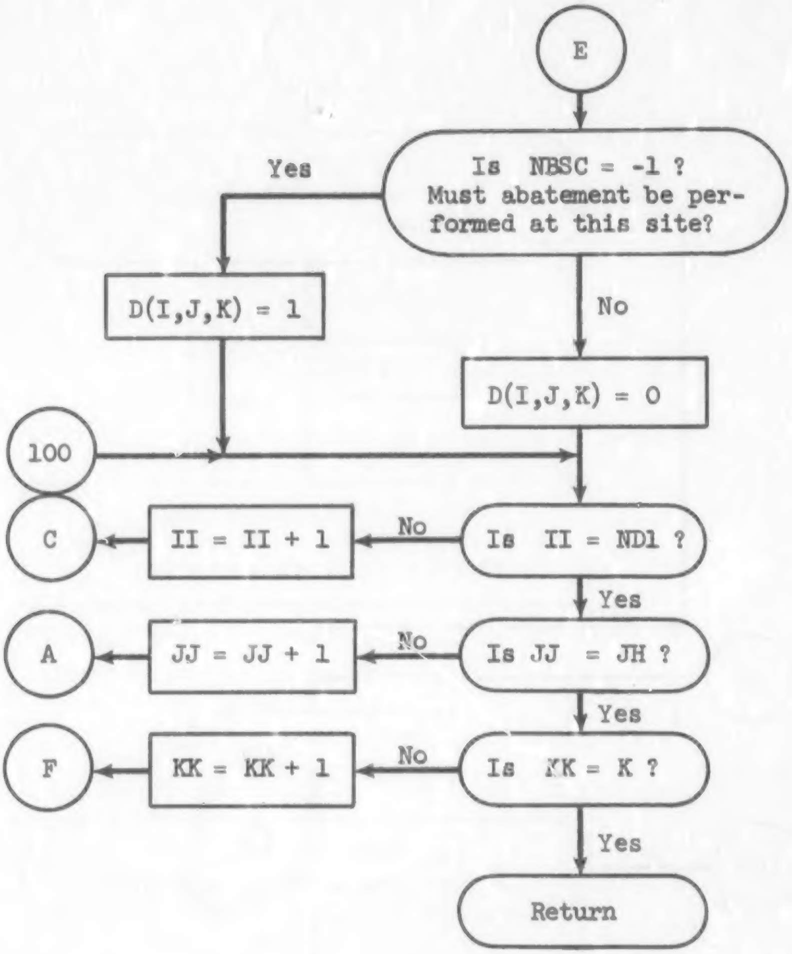

E
Is NBSC = -1 ? Must abatement be performed at this site?
Yes
No
D(I,J,K) = 1
D(I,J,K) = 0
100
C
II = II + 1
No
Is II = ND1 ?
Yes
A
JJ = JJ + 1
No
Is JJ = JH ?
Yes
F
KK = KK + 1
No
Is KK = K ?
Yes
Return

Figure C.25 Subroutine ZO

APPENDIX D

COST OF DRAINAGE TREATMENT

INTRODUCTION

In the allocation of resources for the control of acid mine drainage effects, the preferred allocation is strongly dependent upon the specific treatment and abatement methods allowed, as well as the specific costs involved for each allowable method. At each mine source or instream treatment site, an engineering analysis will choose the cheapest (according to an economic criterion and desired effect on acid flow) treatment or abatement methods. Moreover, the specific treatment or abatement costs will influence the global or basin-wide selection of treatment and/or abatement control actions at each site. Realizing that input cost data will be required at each site to operate the optimization models described in Appendix C, an effort was made to generate cost functions for several treatment and abatement techniques which would be representative of observed costs, have relatively high confidence levels, and be amenable to extrapolation beyond the limits of the data base from which they were derived.

Table D.1 is presented to set the context in which these cost functions must perform. Table D.1 describes, in various systems of units, treatment and abatement cost ranges for a variety of techniques. It should be noted that these data are abstracted from a 1968 publication based on 1951 to 1966 cost figures. These numbers are presented to show the relative cost levels on a per gallon basis for treatment methods and relative cost for abatement methods.

Predesign cost estimates are extremely important for determination of the feasibility and applications of a proposed treatment project. The basic costs which are of interest to the designers are the per gallon, per acre or yearly treatment and abatement costs. To construct parametric relationships for accurately estimating these costs requires large amounts of data, since the total product costs are complex functions of a number of variables and the costs are also changed with locations and time due to different economic conditions. Usually, in

Table D.1. COST AND EFFECTIVENESS OF VARIOUS TECHNIQUES FOR CONTROLLING ACID MINE DRAINAGE

Technique	Effectiveness (% acid removal)	Cost ($)
Treatment		
Neutralization	80 to 97	.10 - 1.3/kgal
Distillation	97 to 99	.40 - 3.25/kgal
Reverse Osmosis	90 to 97	.68 - 2.57/kgal
Ion Exchange	90 to 99	.61 - 2.53/kgal
Freezing	90 to 99	.67 - 3.23/kgal
Electrodialysis	25 to 95	.58 - 2.52/kgal
At-Source Control		
Water Diversion	25 to 75	300 - 2000/acre
Mine Sealing	10 to 80	1000 - 2000/acre
Surface Restoration	25 to 75	300 - 3000/acre
Revegetation	5 to 25	70 - 350/acre

Note: The above table is outlined from J. Martin and R. D. Hill, "Mine Drainage Research Program of the Federal Water Pollution Control Administration," Paper presented at the Second Symposium on Coal Mine Drainage Research, Pittsburg, Pennsylvania, May 14-15, 1968.

order to account for time effects upon costs, time correcting cost indexes should be incorporated.

As suggested by different researchers (12,13), the total treatment process costs should be divided into manufacturing costs and general expenses. In order to evaluate each of the items, the total capital costs should be considered first. For the sake of elucidating the basic principles of the cost estimates, Tables D.2 and D.3 itemize the important costs that should be considered in the economic studies. These two tables are summarized from Reference (12).

TREATMENT COST MODEL

From Table D.1, it could be concluded that neutralization would be the crucial first target for economic modelling. Accordingly, several cost model structures are presented and evaluated for this treatment process. These models are derived from data of the estimated costs of lime

Table D.2. ESTIMATION OF CAPITAL INVESTMENT COST

I. Direct Cost:
Material and labor involved in actual installation of complete facility, about 70 to 85% of the fixed capital investment.

  A. Equipment + Installation + Instrumentation

  B. Building + Process + Auxiliary

  C. Service Facilities + Yard Improvements

II. Indirect Cost:
Expenses which are not directly involved with material and labor of actual installation of complete facility

  A. Engineering and Supervision

  B. Construction expense and Contractor's fee

  C. Contingency.

III. Fixed-capital Investment = Direct Costs + Indirect Costs

IV. Working Capital (10 to 20 percent of total capital investment)

V. Total Capital Investment = Fixed-capital Investment + Working Capital.

Table D.3. ESTIMATION OF TOTAL PRODUCTION COSTS

I. Manufacturing Cost = Direct Production Cost + Fixed Charge + Plant Overhead Costs

   A. Direct Production Costs (about 60% of total product cost)

      1. Raw Material
      2. Operating Labor (10 to 20% of total product cost)
      3. Utilities (10 to 20% of total product cost)
      4. Maintenance and Repairs (2 to 10% of fixed capital investment cost)
      5. Laboratory Charges (10 to 20% of operating labor)

   B. Fixed Charges (10 to 20% of total product cost)

      1. Depreciation (about 10% of fixed capital investment depends on life periods)
      2. Local Tax, Insurance and Rent (about 5 to 10% of fixed-capital investment).

   C. Plant Overhead Costs (about 5 to 15% of total production cost including general plant upkeep and overhead, payroll overhead, packaging, medical services, recreation, salvage and so on.

II. General Expenses = Administrative Costs + Distribution and Selling Costs + Research and Development Costs

III. Total Production Cost = Manufacturing Cost + General Expenses neutralization of acid mine drainage obtained by West Virginia University, Morgantown, West Virginia in 1967. The original data are shown in Table D.4. The total costs are assumed to be the sum of plant costs (except sludge removal), lime costs, labor costs, cost of sludge disposal, maintenance costs, and contingency costs. The cost of hydrated lime is taken at $24/ton bagged, $22/ton bulk. The data which are used for the basis of the model includes the following items:

Description	% of Total Costs
A. Direct Production Costs	
1. Raw material (lime)	25 - 50
2. Labor costs	8 - 12
3. Utilities (for sludge disposal)	8 - 11
4. Maintenance	6 - 8
B. Fixed Charges and Plant Overhead	12 - 30
C. Contingency and Miscellaneous	5 - 12

Because of limitations in the data displayed in Table D.4, the cost model analysis is limited to two independent variables and relatively simple relationships. A reasonable relationship should show total treatment costs on a unit volume basis increasing with decrease in plant size and increase in acidity concentration. Therefore, it was first proposed that:

$$T = \alpha(A)^a(B)^b \qquad (D.1)$$

where A is input acidity concentration in ppm,
B is the plant capacity in gal/day,
T is the treatment cost in cents/kgal, and
α, a, b are constant.

In order to estimate the values for α, a, and b, a plot at log T versus log B while holding A constant was generated. Since four sets of data are given in Table D.4, four different values for the slope b using estimates from a least squares regression analysis, were obtained, and then these values were averaged to estimate a mean value of b. In the same manner by holding B constant, and plotting log T versus log A, a mean value of a was obtained. From the results, Equation (D.1) becomes

$$T = 0.5293\ (A)^{0.7}(B)^{-0.088} \qquad (D.2)$$

Table D.4. ESTIMATED COSTS OF LIME NEUTRALIZATION OF ACID MINE DRAINAGE
(all costs in cents/1000 gallons)

Plant Capacity (gal/day)	Approximate acidity concentration	Approximate iron content	Plant cost (except sludge removal)	Lime**	Labor	Sludge disposal	Maintenance	Contingencies	Total	Sludge accumulation (acre-ft/yr)
300,000	6500	2000	12.0	53	14.0	11.0	6.0	5.0	101	13
900,000	6500	2000	11.2	51	12.6	10.5	5.8	5.0	96	39
2,700,000	6500	2000	10.4	49	11.8	10.5*	5.5	5.0	92	117
8,100,000	6500	2000	9.8	48	11.0	10.5*	5.3	5.0	89	351
300,000	3400	1000	9.5	28.0	10.0	8.0	4.0	3.0	62.5	9.8
900,000	3400	1000	8.5	26.0	5.0	7.0	3.0	3.0	52.5	30.1
2,700,000	3400	1000	7.5	25.5	2.5	7.75*	2.5	3.0	48.5	91.0
8,100,000	3400	1000	7.25	25.5	2.0	7.50*	2.5	3.0	47.74	273.0
300,000	1400	650	9.5	12.9	8.0	4.0	3.0	3.0	34.8	4.9
900,000	1400	650	8.5	11.5	4.0	3.5	2.5	3.0	33.0	15.4
2,700,000	1400	650	7.75	11.0	2.0	3.75*	2.0	3.0	29.5	45.4
8,100,000	1400	650	7.25	11.0	1.6	3.75*	2.0	3.0	28.6	136.5
300,000	650	325	8.5	6.1	6.0	2.0	2.5	2.5	27.60	2.8
900,000	650	325	7.5	5.7	3.0	1.8	2.0	2.5	22.50	7.7
2,700,000	650	325	6.75	5.5	1.8	1.9*	1.5	2.5	19.95	23.1
8,100,000	650	325	6.5	5.5	1.0	1.9*	1.5	2.5	18.90	68.6

*These costs allow for excavating some hard rock
**Cost of hydrated lime taken at $24.00/ton bagged, $22.00/ton bulk

However, the problem of solving for the annual treatment cost is always raised. In order to do so, Equation (D.2) must be converted from cents/kgal to dollars/yr. This procedure follows:

Since T is in ¢/kgal and B is in gal/day,

Hence,

$$G \text{ \$/year} = (\$/100 \text{ cents})(T \text{ cents}/1000 \text{ gal})\cdot(365 \text{ days/year})(B \text{ gal/day})$$

$$G = 365 \cdot 10^{-5}(T)(B) \tag{D.3}$$

Substituting Equation (D.3) into Equation (D.2), we obtained

$$G = 1.9323 \cdot 10^{-3} \cdot (A)^{0.7}(B)^{0.932} \tag{D.4}$$

The comparative results from the data and the model are tabulated in Tables D.5 and D.6. Table D.5 shows the treatment costs expressed in terms of cents per 1000 gallons; whereas Table D.6 indicates the results of the total annual costs for a given size of plant. The deviations between model and actual results are less than 25% of actual values, which is reasonable for predesign cost estimates.

However, in order to pursue a better model, an additional coefficient was incorporated. Several models were evaluated, and the best fit for the entire set was determined to be Equation (D.5).

$$G = 365 \times 10^{-5}\,(36.54 + 0.0322A)(B)^{0.932} \tag{D.5}$$

Tables D.7 and D.8 present comparisons between data and actual results, and these tables show a maximum error in predicting the actual results of 11.6% of actual values.

In comparing the two models represented by Equations (D.4) and (D.5), significant improvement is achieved by using Equation (D.5). The maximum deviation of the predicted results from the data is less than 12% compared to 25% deviation obtained by Equation (D.4). A 12% error in predicting costs before a detailed system design is performed should be acceptable is most applications. However, one might note that these models were derived by finding the best fit to the same data that were used to estimate predictive errors; thus, a new data set may give larger errors. Also, the linear relationship between the annual costs and the acidity seems to fit the data better than using 0.7th power as suggested by Equation (D.4). The linear relationship is strongly supported by Figure D.1 where lime cost is plotted against acid concentration for the data listed in Table D.4.

Table D.5. ESTIMATED COSTS OF LIME NEUTRALIZATION OF ACID MINE DRAINAGE BASED ON EQUATION (D.4)
(all costs in cents/1000 gallons)

Plant capacity (gal/day)	Acidity (ppm)	Data result	Model result	Deviation (%)
300,000	6500	101	104.80	3.76
900,000	6500	96	101.27	5.49
2,700,000	6500	92	90.25	-1.90
8,100,000	6500	89	83.75	-5.90
300,000	3400	62.5	66.59	6.54
900,000	3400	52.5	61.79	17.70
2,700,000	3400	48.5	57.34	18.22
8,100,000	3400	47.75	53.21	11.43
300,000	1400	34.8	35.78	2.82
900,000	1400	33.0	33.21	0.64
2,700,000	1400	39.5	30.81	4.44
8,100,000	1400	28.6	28.56	-0.03
300,000	650	27.6	20.91	-24.2
900,000	650	22.5	19.41	-13.7
2,700,000	650	19.95	18.01	-9.6
8,100,000	650	18.90	16.71	-11.6

Table D.6. ESTIMATED COSTS OF LIME NEUTRALIZATION OF
ACID MINE DRAINAGE BASED ON EQUATION (D.4)
(all costs in dollar per year)

Plant capacity (gal/day)	Acidity (ppm)	Data result ($/yr)	Model result ($/yr)	Deviation (%)
300,000	6500	110,593	114,208	3.30
900,000	6500	315,360	319,497	1.31
2,700,000	6500	906,660	889,495	-1.89
8,100,000	6500	2,631,285	2,476,397	-5.87
300,000	3400	68,437	72,911	6.54
900,000	3400	172,463	202,987	11.08
2,700,000	3400	477,968	565,125	18.20
8,100,000	3400	1,411,729	1.573,337	11.45
300,000	1400	38,106	39,179	2.81
900,000	1400	108,405	109,075	0.62
2,700,000	1400	390,723	303,671	4.45
8,100,000	1400	845,559	843,435	-0.01
300,000	650	30,222	22,898	-24.2
900,000	650	73,913	63,748	-13.75
2,700,000	650	196,607	177,480	-9.73
8,100,000	650	558,779	494,112	-11.37

Table D.7. ESTIMATED COSTS OF LIME NEUTRALIZATION OF ACID MINE DRAINAGE BASED ON EQUATION (D.5)
(all costs in cents/1000 gallons)

Plant capacity (gal/day)	Acidity (ppm)	Data result	Model result	Deviation (%)
300,000	6500	101	104.4	3.26
900,000	6500	96	96.9	0.9
2,700,000	6500	92	89.93	-2.250
8,100,000	6500	89	83.43	-6.23
300,000	3400	62.5	62.02	-0.78
900,000	3400	52.5	57.55	-8.620
2,700,000	3400	48.5	53.40	9.620
8,100,000	3400	47.75	49.53	3.791
300,000	1400	34.8	34.65	+1.59
900,000	1400	33.0	32.16	-2.54
2,700,000	1400	29.5	29.84	+1.15
8,100,000	1400	28.6	27.69	-3.18
300,000	650	27.6	24.40	-11.59
900,000	650	22.5	22.64	0.62
2,700,000	650	19.95	21.00	5.26
8,100,000	650	18.90	19.49	3.12

Table D.8. ESTIMATED COSTS OF LIME NEUTRALIZATION OF ACID MINE DRAINAGE BASED ON EQUATION (D.5)
(all costs in dollars/year)

Plant capacity (gal/day)	Acidity (ppm)	Data result	Model result	Deviation (%)
300,000	6500	110,593	114,323	3.37
900,000	6500	315,360	318,282	0.93
2,700,000	6500	906,660	886,112	-2.23
8,100,000	6500	2,631,285	2,466,979	-6.24
300,000	3400	60,477	67,897	-0.76
900,000	3400	172,463	189,019	9.60
2,700,000	3400	477,968	526,238	10.10
8,100,000	3400	1,411,729	1,465,073	3.75
300,000	1400	38,106	37,938	-0.44
900,000	1400	108,405	105,624	-2.56
2,700,000	1400	290,723	294,062	1.15
8,100,000	1400	845,559	818,685	3.18
300,000	650	30,222	26,708	-11.63
900,000	650	73,913	74,346	0.60
2,700,000	650	196,607	207,010	5.29
8,100,000	650	558,779	576,327	3.14

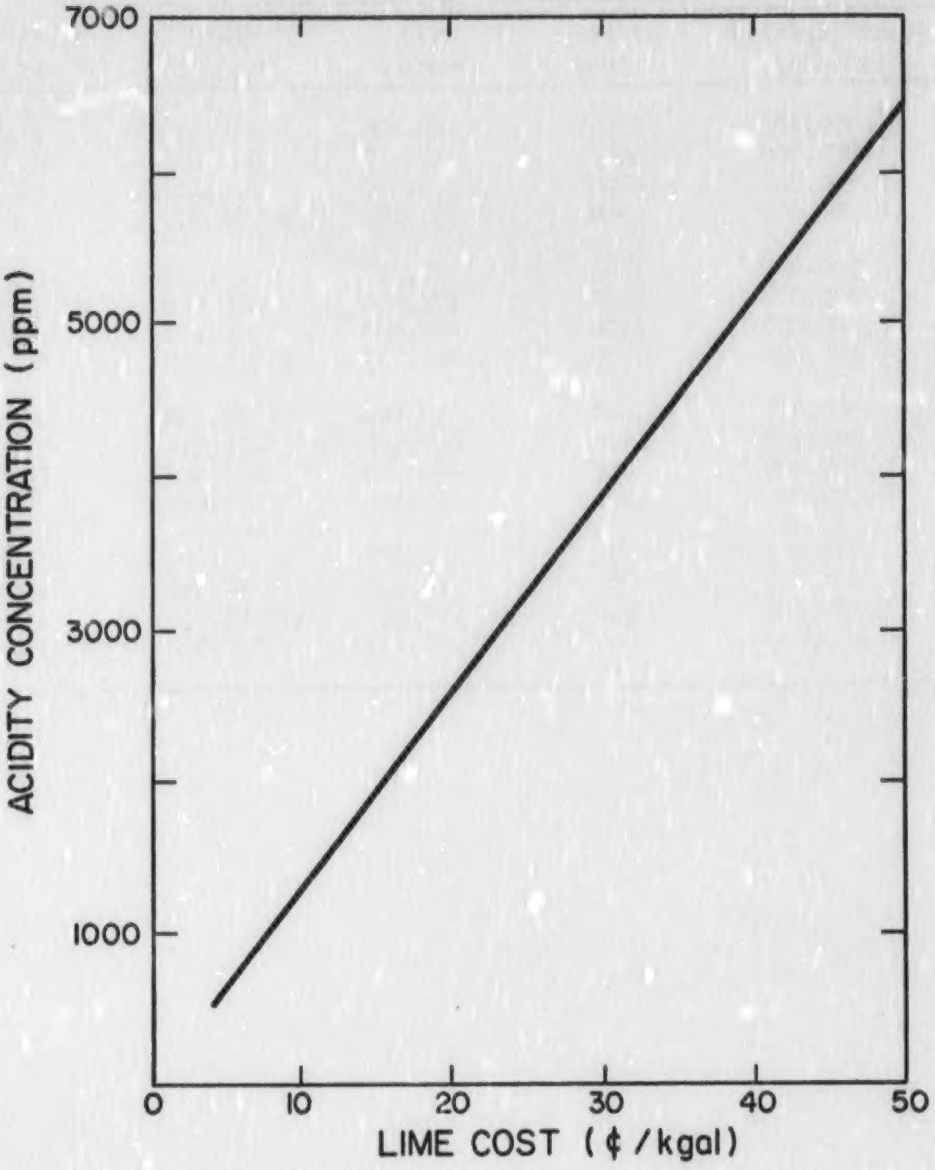


Figure D.1. A plot of acidity concentration vs. lime cost

The annual treatment costs are exponentially increased with plant capacity. At all levels of acidity, the annual costs increase the most rapidly in the data set from 1 x 10^6 to 3 x 10^6 gallons per day.

In order to apply the model represented by Equation (D.5) to arbitrary process environments, the following factors which influence the cost estimates should be considered with great care. As indicated by the data given in Table D.4, the cost of limestone contributes 25 to 50% of the total cost. Limestone costs may be an essential factor affecting the determination of the total costs. Limestone costs vary with location of the treatment plant with respect to the nearest lime producer, and they also change from time to time. In order to obtain a reasonable estimate of the treatment cost, a correction factor for limestone costs dependent on location and time, should be taken into account. Construction costs also vary with location and the type of excavation required, but this factor is hard to predict. Other costs, such as labor costs, plant costs, costs for sludge disposal, maintenance and contingency costs are relatively constant with locations, and each of them contributes only a relatively small percentage to the total costs compared to the limestone costs. Therefore, only the time effect on the construction costs should be considered. This can be done simply by incorporating the cost indexes, which are available in the literature, into the cost estimates.

EXAMPLE APPLICATION

An example application is presented in order to illustrate the application of the cost model and the inclusion of the corrections mentioned above.

Sample Problem

A treatment plant was built in Morgantown, West Virginia, in 1967. This plant could handle 300,000 gallons of water with an acid concentration of 6500 ppm per day. The following problems are posed:

(a) If the same plant was to be used in 1970, what would the annual total cost be?

(b) If a treatment plant which could handle 900,000 gallons of water with an acid concentration of 3400 ppm per day were to be built in Central Ohio, what would the approximate annual cost be in 1970? The total cost for treating 1000 gallons of water with an acid concentration of 3400 ppm in Central Ohio is estimated to be $.30.

Solution to the Problems

Problem (a): Choosing 1967 as the base year, its cost index is assumed to be 100. From the literature, the cost index for 1970 is found to be 114 (this is only a hypothetic value).

The annual cost for the plant can be computed from Equation (D.5):

$$\begin{aligned} G &= 365 \times 10^{-5} (36.54 + 0.0322A)(B)^{0.932} \\ &= 365 \times 10^{-5} (36.54 + 0.0322 \times 6500)(300000)^{0.932} \\ &= \$114,188. \end{aligned}$$

The annual cost for the plant in 1970 would be

$$\begin{aligned} G_{1970} &= G_{1967} \cdot \frac{\text{Cost Index in 1970}}{\text{Cost Index in 1967}} \\ &= 114,188 \times 1.14 \\ &= \$130,174. \end{aligned}$$

Problem (b): From Equation (D.5):

$$\begin{aligned} G &= 365 \times 10^{-5} (36.54 + .0322 \times 3400)(900000)^{0.932} \\ &= \$188,824 \end{aligned}$$

Since the annual cost, G, is the sum of lime cost, labor cost, sludge disposal cost, plant cost, maintenance cost and contingency cost, all these costs are based on the data obtained in West Virginia. In order to correct for the location effect on lime cost, one must calculate:

(1) From Figure D.2, with acid concentration of 3400 ppm, the treatment cost for 1000 gallons of acid water is about \$0.26 in West Virginia; thus, the difference between the total lime costs in West Virginia and Central Ohio is,

$$\begin{aligned} (0.3 - 0.26)(\$/\text{kgal})(365 \text{ day/yr})(900 \text{ kgal/day}) &= 0.04 \times 365 \times 900 \\ &= \$13,150 \end{aligned}$$

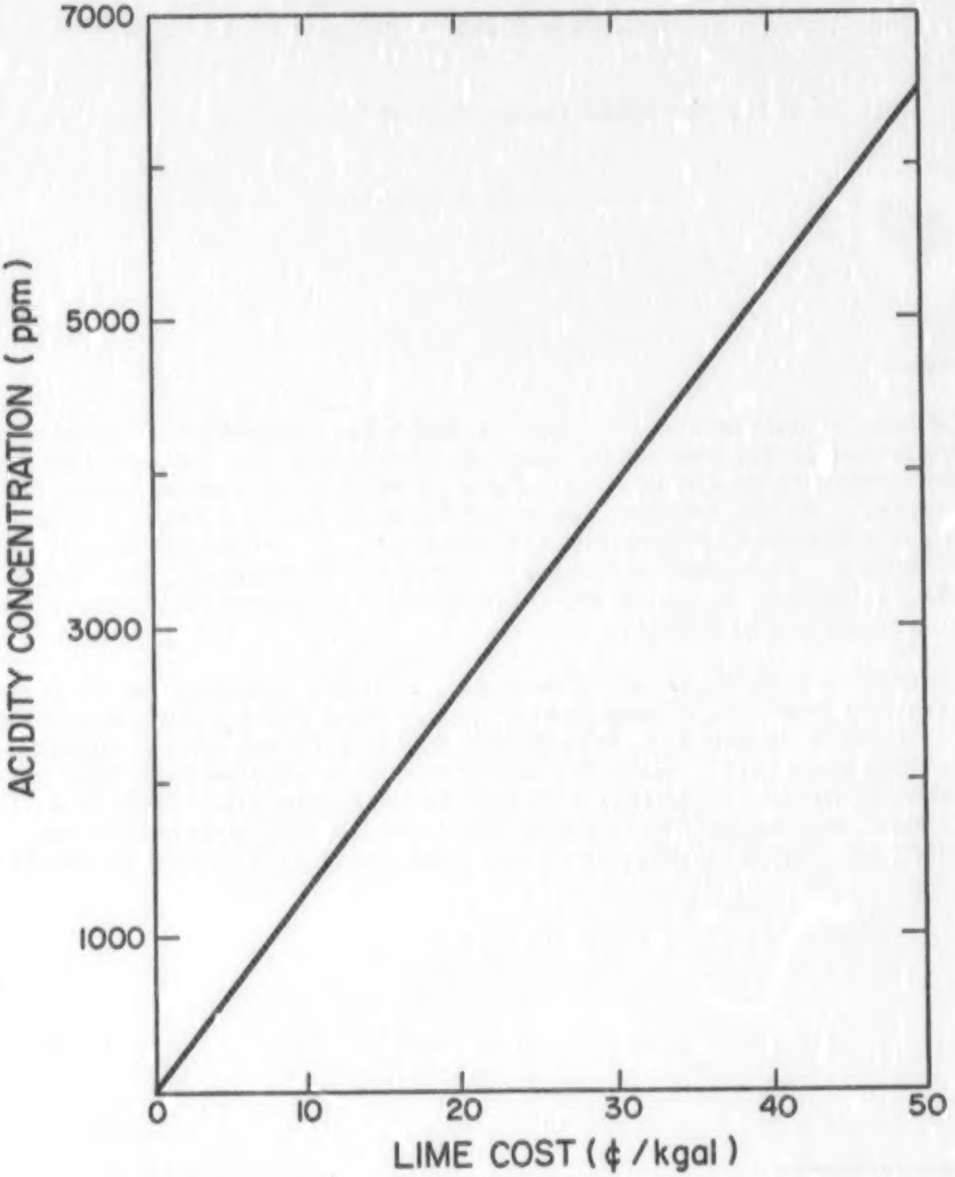


Figure D.2. A plot of acidity concentration vs. lime cost
1967 West Virginia University data

(2) For the treatment plant in Central Ohio, the true annual cost in 1967 should be corrected to

$$188{,}824 + 13{,}150 = 201{,}974^{*}$$

(3) In 1970, the total annual cost would be

$$G = 201{,}974 \times (114/100)$$

$$= 230{,}250$$

SUMMARY

This type of mathematical analysis permits the estimation of treatment costs based on information concerning acidity and plant size. The results predicted by the model should only be used as preliminary cost estimates. As one may note the original data given in Table D.4 do not provide sufficient information for detailed cost estimates that are sensitive to individual situations. However, by utilizing the cost model, designers should be able to determine preferred plant sizes and treatment plant locations.

As pointed out by Peter and Timmerhaus, a "Study Estimate" or "Factored Estimate" based on the knowledge of major items has a probable accuracy of ±30% (12). Based upon this standard, the cost estimating relationship, Equation (D.5), describes the observed results well in that the maximum error in the predicted result is less than 12%. Therefore, this cost estimating relationship should be used by considering how significant factors affect the individual cases in a particular application.

---

*If the lime cost difference between two locations is less than 20%, the correction for location effect is unnecessary. As shown in the above case, the correction term only contributes to less than 5% of the total costs.

REFERENCES FOR APPENDIX D

1. E. Martin and R. Hill, "Mine Drainage Research Program of the Federal Water Pollution Control Administration," paper presented at Second Symposium on Coal Mine Drainage Research, Pittsburgh, Pennsylvania, May 14-15, 1968.

2. C. T. Holland, "Experience in Operating an Experimental Acid Mine Drainage Treatment Plant," 13, No. 2, 157th National Meeting of the American Chemical Society, Minneapolis, Minnesota, April 13-18, 1969.

3. R. D. Hill and R. C. Wilmonth, "Limestone Treatment of AMD," A.I.Ch.E. 250, p. 162.

4. "Treatment of Acid Mine Drainage by Ozone Oxidation," Department of Applied Science, Brookhaven National Lab., Associated University, Inc., Dec., 1972.

5. "Evaluation of a New Acid Mine Drainage Treatment Process," Water Pollution Control Research Series, 14010 DYI 02/71.

6. R. B. Scott, R. Hill and R. Wilmonth, "Cost of Reclaimation and Acid Mine Drainage Abatement," Water Quality Office, Environmental Protection Agency, Cincinnati, Ohio, 43226, Oct., 21-23, 1970.

7. J. B. Jone and S. Ruggeri, "Abatement of Pollution from Abandoned Coal Mines by Means of in situ Precipitation Technique," paper presented at the 157th National Meeting American Chemical Society, 13, No. 2, April 13-18, 1969.

8. R. A. Tybout, "Private and Social Costing and Pricing as Decision Determinants," paper available at The Ohio State University, Department of Economics.

9. Hoggan, et al., "State and Local Capability to Share Financial Responsibility of Water Development with Federal Government," U. S. Water Resources Council, 1971.

10. "West Virginia Water Pollution Control Program," EPA and FWQA Dept., 1970.

11. Peters and Timmerhaus, Plant Design and Economics for Chemical Engineers, McGraw-Hill Book, 2nd edition, 1968.

12. J. P. Gibb, "Cost of Water Treatment in Illinois," Illinois State Water Survey, Urbana, 1968.

13. Morth, "Acid Mine Drainage: A Mathematical Model," Ph.D. Dissertation, The Ohio State University, 1971.

14. Chow, K., "Computer Simulation of Acid Mine Drainage," Master's Thesis, The Ohio State University, 1972.

15. Proc. Second Symposium on Coal Mine Drainage Research, Pittsburgh, 1968.

16. Water Pollution Control Research Series, 14010, DYG, EPA, Aug., 1972.

17. Water Pollution Control Research Series, 14010, FQR, EPA, March, 1972.

18. Water Pollution Control Research Series, 14010, DYK, EPA, March, 1972.

19. Water Pollution Control Research Series, 12010, EQF, EPA, March, 1971.

20. Water Pollution Control Research Series, 14010, FNQ, EPA, February, 1972.

21. Lund, H., Industrial Pollution Control Handbook, McGraw-Hill, 1973.

22. Environmental Protection Technology Series, EPA* R2-72-056, Nov., 1972.

23. Water Pollution Control Research Series, 14010, EQF, EPA, February, 1971.

24. Water Pollution Control Research Series, 12010, FNQ, EPA, March, 1970.